

Ross scheme, Newton–Raphson iterative methods and time-stepping strategies for solving the mixed form of Richards' equation

Fadji Hassane Maina, Philippe Ackerer

▶ **To cite this version:**

Fadji Hassane Maina, Philippe Ackerer. Ross scheme, Newton–Raphson iterative methods and time-stepping strategies for solving the mixed form of Richards' equation. Hydrology and Earth System Sciences, 2017, 21, pp.2667 - 2683. 10.5194/hess-21-2667-2017 . hal-03516940

HAL Id: hal-03516940
https://cnrs.hal.science/hal-03516940

Submitted on 7 Jan 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Hydrol. Earth Syst. Sci., 21, 2667–2683, 2017
https://doi.org/10.5194/hess-21-2667-2017
© Author(s) 2017. This work is distributed under
the Creative Commons Attribution 3.0 License.

Hydrology and Earth System Sciences
Open Access
EGU

Ross scheme, Newton–Raphson iterative methods and time-stepping strategies for solving the mixed form of Richards' equation

Fadji Hassane Maina[1,2] **and Philippe Ackerer**[1]

[1]Laboratoire d'Hydrologie et de Géochimie de Strasbourg, Univ. Strasbourg/EOST – CNRS, 1 rue Blessig, 67084 Strasbourg, France
[2]CEA-Laboratoire de Modélisation des Transferts dans l'Environnement, Bât. 225, 13108 Saint Paul lez Durance cedex, France

Correspondence to: Philippe Ackerer (ackerer@unistra.fr)

Received: 23 November 2016 – Discussion started: 20 December 2016
Revised: 5 May 2017 – Accepted: 8 May 2017 – Published: 8 June 2017

Abstract. The solution of the mathematical model for flow in variably saturated porous media described by the Richards equation (RE) is subject to heavy numerical difficulties due to its highly nonlinear properties and remains very challenging. Two different algorithms are used in this work to solve the mixed form of RE: the traditional iterative algorithm and a time-adaptive algorithm consisting of changing the time-step magnitude within the iteration procedure while the nonlinear parameters are computed with the state variable at the previous time. The Ross method is an example of this type of scheme, and we show that it is equivalent to the Newton–Raphson method with a time-adaptive algorithm.

Both algorithms are coupled to different time-stepping strategies: the standard heuristic approach based on the number of iterations and two strategies based on the time truncation error or on the change in water saturation. Three different test cases are used to evaluate the efficiency of these algorithms.

The numerical results highlight the necessity of implementing an estimate of the time truncation errors.

1 Introduction

Water movement in soils is one of the key processes in the water cycle since it contributes to the renewal of groundwater resources through recharge, to vegetation growth through transpiration, to soil fertility through salinization/alteration and to atmospheric humidity through evaporation and transpiration. Water movement is usually modeled using the Richards equation (Richards, 1931), which is now commonly adopted for many studies in soil science and/or hydrology, including the use of physically based hydrological models applied to large-scale catchments and for long time simulations (e.g., for climate change studies). However, this equation is highly nonlinear, and despite numerous efforts over the last 40 years, its numerical solution requires much computational time.

Assuming a rigid solid matrix, the Richards equation (RE) is given by

$$\begin{cases} \dfrac{\partial \theta}{\partial t} + S_{\mathrm{w}} s_0 \dfrac{\partial \psi}{\partial t} + \nabla \cdot \boldsymbol{q} = f \\ \boldsymbol{q} = -k_r(\psi)\boldsymbol{K}\left[\nabla \psi + \nabla z\right] \end{cases} \tag{1}$$

where θ is the volumetric water content ($\mathrm{L}^3\,\mathrm{L}^{-3}$), S_{w} is the water saturation (–), s_0 accounts for fluid compressibility (L^{-1}), ψ is the pressure head (L), q is the water flux based on the extended Darcy's law ($\mathrm{L\,T}^{-1}$), t is the time (T), z is the vertical coordinate (positive upward) (L), f is the sink/source term (T^{-1}), K is the saturated hydraulic conductivity tensor ($\mathrm{L\,T}^{-1}$) and $k_r(\psi)$ is the relative hydraulic conductivity (−). The model includes initial and boundary conditions of the Dirichlet (prescribed pressure head) or Neumann (prescribed flux) type.

Equation (1) is also called the mixed form of RE. Two alternative formulations of the mixed form exist for RE.

The pressure form is defined by

$$\begin{cases} \left[C(\psi) + S_{\mathrm{w}} s_0\right] \dfrac{\partial \psi}{\partial t} + \nabla \cdot \boldsymbol{q} = f \\ \boldsymbol{q} = -k_r(\psi)\mathbf{K}\left[\nabla \psi + \nabla z\right] \end{cases} \tag{2}$$

where $C(\psi) = \frac{\partial\theta}{\partial\psi}$ is the specific moisture capacity (L^{-1}), and the soil moisture form that is restricted to unsaturated conditions is defined by

$$\begin{cases} \dfrac{\partial\theta}{\partial t} + \nabla\cdot\boldsymbol{q} = f \\ \boldsymbol{q} = -\left(\mathbf{D}(\theta)\nabla\theta + k_r(\theta)\mathbf{K}\nabla z\right) \end{cases} \tag{3}$$

where $\mathbf{D}(\theta) = k_r(\theta)\mathbf{K}\frac{d\psi}{d\theta}$ is the pore water diffusivity ($L^2\,T^{-1}$).

Constitutive relations are required to solve RE. For the pressure–water content relationship, the most common model is the Van Genuchten model (van Genuchten, 1980):

$$S_w(\psi) = \frac{\theta(\psi)-\theta_r}{\theta_s-\theta_r} = \begin{cases} (1+|\alpha\psi|^{\eta})^{-m}\psi < 0, \\ 1\psi \geq 0, \end{cases} \tag{4}$$

where $m = 1-1/\eta$, S_w is the effective saturation, θ_r and θ_s are the residual and saturated volumetric water content, respectively, and α and η are experimentally estimated coefficients.

This model is usually associated with the Mualem model (Mualem, 1976) for the relative permeability of the aqueous phase:

$$k_r(S_w) = \begin{cases} S_w^{1/2}\left[1-\left(1-S_w^{1/m}\right)^m\right]^2\psi < 0, \\ 1.0\psi \geq 0. \end{cases} \tag{5}$$

A summary of the most popular relations can be found in Belfort et al. (2013).

Due to the strong heterogeneities of the unsaturated zone and nonlinearities in the constitutive relations (Eqs. 4 and 5), analytical solution of RE does not exist except in special cases (Celia et al., 1990; van Dam and Feddes, 2000). Therefore, numerical methods such as finite difference (Feddes et al., 1988; Romano et al., 1998; van Dam and Feddes, 2000), finite element (Gottardi and Venutelli, 2001), and mixed finite element (Bause and Knabner, 2004; Bergamaschi and Putti, 1999; Fahs et al., 2009; Farthing et al., 2003) are used to solve RE.

Iterative methods based on the Picard (fixed point) or Newton–Raphson approach (Lehmann and Ackerer, 1998; Paniconi and Putti, 1994) are the most popular techniques for solving this highly nonlinear equation. Alternative iterative methods are based on transform formulations (Crevoisier et al., 2009; Ross and Bristow, 1990; Williams et al., 2000; Zha et al., 2013) or the method of lines (Fahs et al., 2009; Matthews et al., 2004; Miller et al., 1998; Tocci et al., 1997).

Adaptive time-stepping strategies based on time truncation error control were found to be superior to other approaches (Hirthe and Graf, 2012; Kavetski et al., 2001; Tocci et al., 1997). The method of lines using the DASPK integrator was applied to the Richards' equation by Matthews et al. (2004), Miller et al. (1998), and Tocci et al. (1997), among others. The method of lines consists of discretization of the spatial part of the PDE only, leading to a system of ordinary differential equations. It has been found to be significantly more efficient than other temporal discretizations (Miller et al., 2006). However, Kavetski and Binning (2002b) reported difficulties in obtaining convergence for the DASPK solver associated with an arithmetic mean of inter-block conductivities for the most difficult problem addressed by Miller et al. (1998). Additionally, very few non-iterative schemes have been developed (Kavetski and Binning, 2004, 2002a; Paniconi et al., 1991).

Despite the many existing numerical methods, solution of RE is still a challenging research topic, with many remaining questions about reduction of the computational time, treatment of nonlinearities, and improvement of the accuracy of these methods for difficult problems such as infiltration in very dry soils (Diersch and Perrochet, 1999; Forsyth et al., 1995; Hills, 1989). The need for efficient algorithms to solve this equation has increased during recent decades because it has been recognized that explicit modeling of flow in the unsaturated zone has to be implemented in land surface models (Vergnes et al., 2012). In their recent review of land surface models, Clarke et al. (2015) push for a mechanistic modeling of the flow in soils. They consider the implementation of the mixed form of the Richards equation to be an improvement to the modeling of soil moisture variations. They also underline the need for efficient algorithms to solve RE, to allow the implementation of stochastic approaches and/or automatic parameter estimations.

In this study, we analyzed the performance of different algorithms based on the Newton–Raphson method since the classical Picard scheme has been found to be less efficient (Lehmann and Ackerer, 1998). Applied to the soil moisture form of RE, we demonstrate that the recently developed Ross method (Ross, 2003; Crevoisier et al., 2009; Zha et al., 2013) is equivalent to the Newton–Raphson method (Sect. 2). A detailed presentation of the Newton–Raphson method applied to the mixed form or RE is given in Sect. 3. The standard Newton–Raphson algorithm is based on the computation of the corresponding matrices in an iterative way by updating the parameters until convergence. An alternative algorithm has been suggested more recently where the parameters are kept unchanged within one time step and the time step is adapted to reach convergence. This algorithm has been applied to the pressure-based form of RE by Kavetski and Binning (2002a) and to the soil moisture form by Crevoisier et al. (2009), Ross (2003), and Zha et al. (2013). Although this algorithm is called "non-iterative" because the parameters are not updated during the calculation, iterations may be necessary to adapt the magnitude of the time step. Therefore, in the following, we will refer to the usual algorithm as "iterative" and to the alternative algorithm as "time-adaptive". To our knowledge, this alternative algorithm has never been applied to the mixed form of RE. Section 4 is dedicated to both algorithms and to the time-stepping strategy used for solving RE. Finally, in Sect. 5, the numerical accuracy and robust-

ness of the algorithms applied to the mixed form of RE are evaluated using three different test cases.

2 The Ross method and the Newton–Raphson method

The moisture-based formulation is applicable in unsaturated conditions only and is prone to numerical difficulties in the case of heterogeneous soils, explaining the reduced attention directed to this formulation. However, discontinuous water content can be handled by adapted schemes, and the moisture-based formulation appears to be very accurate for initially dry conditions (Zha et al., 2013, 2015).

Ross (2003) suggested a non-iterative formulation that has been recently extended to different soil conditions (Crevoisier et al., 2009; Varado et al., 2006a) and to two and three dimensions (Zha et al., 2013).

In its initial one-dimensional finite-volume formulation and for a volume (cell) i, the Ross method (Ross, 2003) is based on the following set of equations:

$$\begin{aligned}\frac{\Delta z}{\Delta t}\left(\theta_i^{n+1}-\theta_i^n\right) &= \frac{\Delta z}{\Delta t}\left(\theta_{s,i}-\theta_{r,i}\right)\left(S_i^{n+1}-S_i^n\right)\\ &= q_-^\sigma - q_+^\sigma\end{aligned} \tag{6}$$

with

$$\begin{cases} q_+^\sigma = q_+^n + \sigma\left[\left(\frac{\partial q_i^n}{\partial S_i^n}\right)\left(S_i^{n+1}-S_i^n\right)+\left(\frac{\partial q_i^n}{\partial S_{i+1}^n}\right)\left(S_{i+1}^{n+1}-S_{i+1}^n\right)\right] \\ q_-^\sigma = q_-^n + \sigma\left[\left(\frac{\partial q_i^n}{\partial S_i^n}\right)\left(S_i^{n+1}-S_i^n\right)+\left(\frac{\partial q_i^n}{\partial S_{i-1}^n}\right)\left(S_{i-1}^{n+1}-S_{i-1}^n\right)\right], \end{cases} \tag{7}$$

where S_i^{n+1} is the water saturation at cell/node i at time $(n+1)$, q_-^σ (or q_+^σ) is the water flux between cell i and $(i-1)$ (or $i+1$) at time $t = t^n + \sigma\ \Delta t, \sigma \in [0,1]$ and Δz is the size of cell i. $\theta_{s,i}$ is the saturated water content and $\theta_{r,i}$ is the residual water content. For simplicity, we assume here that all cells are of the same size.

The previous mass balance Eq. (6) leads to the following equation for cell i:

$$\begin{aligned}&-\left(\frac{\partial q_-^n}{\partial S_{i-1}^n}\right)\left(S_{i-1}^{n+1}-S_{i-1}^n\right)\\ &+\left[\frac{\Delta z}{\sigma\,\Delta t}\left(\theta_{s,i}-\theta_{r,i}\right)-\left(\left(\frac{\partial q_-^n}{\partial S_i^n}\right)-\left(\frac{\partial q_+^n}{\partial S_i^n}\right)\right)\right]\\ &\left(S_i^{n+1}-S_i^n\right)\\ &+\left(\frac{\partial q_+^n}{\partial S_{i+1}^n}\right)\left(S_{i+1}^{n+1}-S_{i+1}^n\right)=q_-^n-q_+^n.\end{aligned} \tag{8}$$

The Newton–Raphson method was initially developed as a root-finding algorithm of an arbitrary equation that has been generalized for solving a system of nonlinear equations. Applied to the soil moisture form of RE and using an implicit scheme, the NR consists in defining a residual based on the mass balance equation (Eq. 6) at iteration k for time step $n+1$ and for cell i written as

$$\boldsymbol{R}_i^{n+1,k} = \frac{\Delta z}{\Delta t}\left(\theta_{s,i}-\theta_{r,i}\right)\left(S_i^{n+1,k}-S_i^n\right)+q_+^{n+1,k}-q_-^{n+1,k}, \tag{9}$$

where $\boldsymbol{R}_i^{n+1,k}$ is called the residual.

The NR consists in computing the solution at iteration $k+1$ by estimating the residual of the next iteration $\boldsymbol{R}_i^{n+1,k+1}$ using a first-order Taylor development and setting it equal to zero as

$$\frac{\boldsymbol{R}_i^{n+1,k}}{\partial S^{n+1,k}}\left(S_i^{n+1,k+1}-S_i^{n+1,k}\right)+\boldsymbol{R}_i^{n+1,k}=0. \tag{10}$$

The derivatives of this residual are

$$\begin{cases} \frac{\partial \boldsymbol{R}_i^{n+1,k}}{\partial S_{i-1}^{n+1,k}} = -\frac{\partial q_-^{n+1,k}}{\partial S_{i-1}^{n+1,k}}, \\ \frac{\partial \boldsymbol{R}_i^{n+1,k}}{\partial S_i^{n+1,k}} = \frac{\Delta z}{\Delta t}\left(\theta_{s,i}-\theta_{r,i}\right)+\frac{\partial q_+^{n+1,k}}{\partial S_i^{n+1,k}}-\frac{\partial q_-^{n+1,k}}{\partial S_i^{n+1,k}}, \\ \frac{\partial \boldsymbol{R}_i^{n+1,k}}{\partial S_{i+1}^{n+1,k}} = \frac{\partial q_+^{n+1,k}}{\partial S_{i+1}^{n+1,k}}, \end{cases} \tag{11}$$

which leads to the following set of linear equations:

$$\begin{aligned}&-\frac{\partial q_-^{n+1,k}}{\partial S_{i-1}^{n+1,k}}\left(S_{i-1}^{n+1,k+1}-S_{i-1}^{n+1,k}\right)\\ &+\left[\frac{\Delta z}{\Delta t}\left(\theta_{s,i}-\theta_{r,i}\right)+\frac{\partial q_+^{n+1,k}}{\partial S_i^{n+1,k}}-\frac{\partial q_-^{n+1,k}}{\partial S_i^{n+1,k}}\right]\\ &\left(S_i^{n+1,k+1}-S_i^{n+1,k}\right)\\ &+\frac{\partial q_+^{n+1,k}}{\partial S_{i+1}^{n+1,k}}\left(S_{i+1}^{n+1,k+1}-S_{i+1}^{n+1,k}\right)=\frac{\Delta z}{\Delta t}\left(\theta_{s,i}-\theta_{r,i}\right)\\ &\left(S_i^{n+1,k}-S_i^n\right)+q_+^{n+1,k}-q_-^{n+1,k}.\end{aligned} \tag{12}$$

For the first iteration, we have $S_i^{n+1,k+1}=S_i^{n+1}$ and $S_i^{n+1,k}=S_i^n$, and therefore

$$\begin{aligned}&-\frac{\partial q_-^n}{\partial S_{i-1}^n}\left(S_{i-1}^{n+1}-S_{i-1}^n\right)+\left[\frac{\Delta z}{\Delta t}\left(\theta_{s,i}-\theta_{r,i}\right)+\frac{\partial q_+^n}{\partial S_i^n}-\frac{\partial q_-^n}{\partial S_i^n}\right]\\ &\left(S_i^{n+1}-S_i^n\right)\\ &+\frac{\partial q_+^n}{\partial S_{i+1}^n}\left(S_{i+1}^{n+1}-S_{i+1}^n\right)=q_+^n-q_-^n.\end{aligned} \tag{13}$$

Whatever the formulation of the fluxes q (as a function of the pressure (see Eq. A1) or the water content, expressed by Kirchhoff transform as in Ross, 2003, or not), the implicit Ross method (Eq. 8 with $\sigma=1$) is equivalent to the first iteration of the Newton–Raphson method (Eq. 13).

3 Newton–Raphson method for the mixed form Richards' equation

Because the pressure-based formulation does not ensure mass conservation – except for the approximation provided by Rathfelder and Abriola (1994) – and due to the limitations of the moisture-based formulation (see previous section), the mixed formulation has been widely used since the work of Celia et al. (1990).

The mixed form of the Richards equation given by Eq. (1) is rewritten as

$$\frac{\partial\theta}{\partial t} + S_w s_0 \frac{\partial\psi}{\partial t} = \nabla \cdot k_r(\psi)\mathbf{K}[\nabla\psi + \nabla z] + f \tag{14}$$

and is discretized by

$$\mathbf{A}^{n+1,k}\boldsymbol{\psi}^{n+1,k+1} + \mathbf{B}^{n+1,k}\frac{\boldsymbol{\psi}^{n+1,k+1} - \boldsymbol{\psi}^{n}}{\Delta t^{n+1}} + \mathbf{E}\frac{\boldsymbol{\theta}^{n+1,k+1} - \boldsymbol{\theta}^{n}}{\Delta t^{n+1}} = \boldsymbol{F}^{n+1,k}, \tag{15}$$

where $\mathbf{A}$ is the discretized form of the divergence term, $\mathbf{B}$ and $\mathbf{E}$ are the discretized forms of the storage terms, $\boldsymbol{F}$ is the discretized form of the sink/source term and the boundary conditions, n is the time step and k is the iteration counter. Δt^{n+1} is the time-step magnitude defined by $\Delta t^{n+1} = t^{n+1} - t^{n}$. Matrices $\mathbf{A}$, $\mathbf{B}$, and $\mathbf{E}$ and vector $\boldsymbol{F}$ depend on the numerical scheme used for the spatial discretization. The implicit scheme is applied for the spatial discretization.

For the Newton–Raphson method, the residual is defined now by

$$\boldsymbol{R}(\boldsymbol{\psi}^{n+1,k}) = \mathbf{A}^{n+1,k}\boldsymbol{\psi}^{n+1,k} + \mathbf{B}^{n+1,k}\frac{\boldsymbol{\psi}^{n+1,k} - \boldsymbol{\psi}^{n}}{\Delta t^{n+1}} + \mathbf{E}\frac{\boldsymbol{\theta}^{n+1,k} - \boldsymbol{\theta}^{n}}{\Delta t^{n+1}} - \boldsymbol{F}^{n+1,k} \tag{16}$$

and its derivatives are

$$\mathbf{R}'(\boldsymbol{\psi}^{n+1,k}) = \mathbf{A}^{n+1,k} + \frac{\partial\mathbf{A}^{n+1,k}}{\partial\boldsymbol{\psi}^{n+1,k}}\boldsymbol{\psi}^{n+1,k} + \frac{\mathbf{B}^{n+1,k}}{\Delta t^{n+1}} + \frac{\partial\mathbf{B}^{n+1,k}}{\partial\boldsymbol{\psi}^{n+1,k}}\frac{\boldsymbol{\psi}^{n+1,k} - \boldsymbol{\psi}^{n}}{\Delta t^{n+1}} + \frac{\mathbf{E}}{\Delta t^{n+1}}\frac{\partial\boldsymbol{\theta}^{n+1,k}}{\partial\boldsymbol{\psi}^{n+1,k}} - \frac{\partial\boldsymbol{F}^{n+1,k}}{\partial\boldsymbol{\psi}^{n+1,k}}. \tag{17}$$

Looking for $\boldsymbol{\psi}^{n+1,k+1}$ such as $\boldsymbol{R}(\boldsymbol{\psi}^{n+1,k+1}) = 0$, the system to be solved is similar to Eq. (10):

$$\mathbf{R}'(\boldsymbol{\psi}^{n+1,k})\Delta\boldsymbol{\psi}^{n+1,k+1} = -\boldsymbol{R}(\boldsymbol{\psi}^{n+1,k}), \tag{18}$$

with $\Delta\boldsymbol{\psi}^{n+1,k+1} = \boldsymbol{\psi}^{n+1,k+1} - \boldsymbol{\psi}^{n+1,k}$.

The NR formulation is also used for the non-iterative scheme by applying only one NR step per time step, with $\boldsymbol{\psi}^{n+1} = \boldsymbol{\psi}^{n+1,1}$ where $\boldsymbol{\psi}^{n+1,0} = \boldsymbol{\psi}^{n}$ (Paniconi et al., 1991; Zha et al., 2015).

4 Algorithms and time-stepping strategy

The usual algorithm used to solve RE consists in defining a time step that remains constant and in iteratively computing the parameters and variables in the following way.

For a given time step n Define the time-step length Δt^{n+1} depending on the time-stepping strategy.

Initialization of the iterative process by setting $\boldsymbol{\psi}^{n+1,1} = \boldsymbol{\psi}^{n}$.

do k = 1, maxit

1. Computation of the variable $\boldsymbol{\theta}^{n+1,k}$, the parameter $\boldsymbol{K}^{n+1,k}$ and their derivatives $\frac{\mathrm{d}\boldsymbol{\theta}^{n+1,k}}{\mathrm{d}\boldsymbol{\psi}^{n+1,k}}$, $\frac{\partial\boldsymbol{K}^{n+1,k}}{\partial\boldsymbol{\psi}^{n+1,k}}$ using $\boldsymbol{\psi}^{n+1,k}$.
2. Computation of the system matrix $\mathbf{R}'$ and the residual $\boldsymbol{R}$.
3. Computation of the system solution $\boldsymbol{\psi}^{n+1,k+1}$.
4. Check convergence. If convergence is achieved, exit.

enddo

Next time step where k is the iteration counter and *maxit* the maximum number of iterations.

The time-adaptive algorithm consists in calculating the nonlinear parameters with the pressure heads computed at time step n and adapting the time-step length. The algorithm is described by the following.

For a given time step n Computation of the variable $\boldsymbol{\theta}^{n}$, the parameter $\boldsymbol{K}^{n}$ and their derivatives $\frac{\mathrm{d}\boldsymbol{\theta}^{n}}{\mathrm{d}\boldsymbol{\psi}^{n}}$, $\frac{\partial\boldsymbol{K}^{n}}{\partial\boldsymbol{\psi}^{n}}$ using $\boldsymbol{\psi}^{n}$.

do k = 1, maxit

1. Define a time step $\Delta t^{n+1,k}$ depending on the time-stepping strategy.
2. Computation of the system matrix $\mathbf{R}'$ and the residual $\boldsymbol{R}$.
3. Computation of the system solution $\boldsymbol{\psi}^{n+1,k+1}$.
4. Check convergence. If convergence is achieved, exit.

enddo

Next time step The main advantage of the alternative algorithm is its avoidance of the computation of the variable $\boldsymbol{\theta}$, the parameter $\boldsymbol{K}$ and their derivatives $\frac{\mathrm{d}\boldsymbol{\theta}}{\mathrm{d}\boldsymbol{\psi}}$ and $\frac{\partial\boldsymbol{K}}{\partial\boldsymbol{\psi}}$ during the iterations. Due to the highly nonlinear relations between $\boldsymbol{\theta}$, $\boldsymbol{K}$, $\frac{\mathrm{d}\boldsymbol{\theta}}{\mathrm{d}\boldsymbol{\psi}}$, $\frac{\partial\boldsymbol{K}}{\partial\boldsymbol{\psi}}$ and the pressure, this computation may require significant CPU time.

The most popular time-step management during the simulation is that of the heuristic type (Miller et al., 2006). The

time step Δt^{n+1} is computed depending on Δt^n and the number of iterations k necessary to reach convergence in the following way:

$$\begin{cases} \text{if } k \le m_1 \ \Delta t^{n+1} = k_1 \Delta t^n k_1 > 1.0, \\ \quad \text{if } m_1 \le k \le m_2 \ \Delta t^{n+1} = \Delta t^n, \\ \text{if } m_2 \le k \ \Delta t^{n+1} = k_2 \Delta t^n k_2 < 1.0, \end{cases} \tag{19}$$

where k_1, k_2, m_1, and m_2 are user-defined constants.

Other heuristic time-step management procedures have been suggested by Kirkland et al. (1992) based on the water volumes exchanged between the adjacent cells of the grid, and by Ross (2003), where the time-step size is controlled by the maximum allowed change in the saturation.

For the Ross method, the fluxes are computed first and the time-step magnitude is calculated accordingly using

$$\Delta t^{n+1} = \frac{\Delta S_{\max}}{\max_i \left(\frac{\left| q^n_{-,i} - q^n_{+,i} \right|}{\Delta z_i (\theta_{s,i} - \theta_{r,i})} \right)} \tag{20}$$

where $\Delta S_{\max}$ is the user-defined maximum allowed saturation change. After the computation of the change in the saturation ΔS, the time step is modified if the maximum of the computed change exceeds $(1+\lambda)\max_i(|\Delta S_i|)$, where λ is a user-defined value, according to

$$\Delta t^{n+1,k} = \frac{\Delta S_{\max}}{\max_i (|\Delta S_i|)} \Delta t^{n+1,k-1}, \tag{21}$$

and the system of equations is solved again. More details about handling the fluxes at boundaries and saturated conditions can be found in Crevoisier et al. (2009), Ross (2003) and Varado et al. (2006b).

The adaptive scheme used in this work evaluates the time steps through truncation error due to the temporal discretization as proposed by Thomas and Gladwell (1988). This scheme was already applied to the pressure-based formulation by Kavetski et al. (2001) and to the moisture-based formulation by Kavetski and Binning (2004).

The difference between the first-order and second-order time approximations can be considered as an estimate of the local truncation error of the first-order scheme. The first-order approximation is given by

$$\boldsymbol{\psi}_1^{n+1} = \boldsymbol{\psi}^n + \Delta t^{n+1} \frac{\partial \boldsymbol{\psi}^n}{\partial t}. \tag{22}$$

The second-order approximation is

$$\begin{aligned} \boldsymbol{\psi}_2^{n+1} &= \boldsymbol{\psi}^n + \Delta t^{n+1} \frac{\partial \boldsymbol{\psi}^n}{\partial t} + \frac{1}{2} \left(\Delta t^{n+1} \right)^2 \frac{\partial^2 \boldsymbol{\psi}^n}{\partial t^2} \\ &= \boldsymbol{\psi}^n + \frac{1}{2} \left(\Delta t^{n+1} \right) \left[\frac{\partial \boldsymbol{\psi}^{n+1}}{\partial t} + \frac{\partial \boldsymbol{\psi}^n}{\partial t} \right] \end{aligned} \tag{23}$$

using $\frac{\partial \boldsymbol{\psi}^{n+1}}{\partial t} = \frac{\partial \boldsymbol{\psi}^n}{\partial t} + \Delta t^{n+1} \frac{\partial^2 \boldsymbol{\psi}^n}{\partial t^2}$.

This truncation error is given by

$$\begin{aligned} \varepsilon_t^{n+1} &= \max_i \left| \psi_{2,i}^{n+1} - \psi_{1,i}^{n+1} \right| \\ &= \frac{1}{2} \Delta t^{n+1} \max_i \left| \frac{\partial \psi_i^{n+1}}{\partial t} - \frac{\partial \psi_i^n}{\partial t} \right| \\ &\approx \frac{1}{2} \Delta t^{n+1} \max_i \left| \frac{\psi_i^{n+1} - \psi_i^n}{\Delta t^{n+1}} - \frac{\psi_i^n - \psi_i^{n-1}}{\Delta t^n} \right|, \end{aligned} \tag{24}$$

when the truncation error is smaller than γ, the temporal truncation error tolerance defined by the user, and the size of the next time step calculated by

$$\Delta t^{n+1} = \Delta t^n \min \left(s \sqrt{\frac{\gamma}{\max(\varepsilon_t^{n+1}, \mathrm{EPS})}}, r_{\max} \right). \tag{25}$$

When the truncation error is larger than γ, the computation is repeated with a reduced time step defined as follows:

$$\Delta t^n = \Delta t^n \max \left(s \sqrt{\frac{\gamma}{\max(\varepsilon_t^{n+1}, \mathrm{EPS})}}, r_{\min} \right), \tag{26}$$

where $r_{\max}$ and $r_{\min}$ are user-defined constants used to avoid overly drastic changes in the time step. s is considered to be a safety factor that ensures that the time-step changes are reasonable. EPS is used to avoid floating point errors when the truncation error becomes too small.

5 Evaluation of the algorithms' performance

We applied the NR method to the mixed form of RE using the standard iterative algorithm and the time-adaptive algorithm. A cell centered finite-volume scheme for the spatial discretization with an implicit Euler scheme for the temporal discretization has been used to solve the partial differential equation and arithmetic means are used to compute the inter-block hydraulic conductivity. The detailed discretizations of the matrix $\mathbf{R}'(\boldsymbol{\psi}^{n+1,k})$ and the vector $\boldsymbol{R}(\boldsymbol{\psi}^{n+1,k})$ (see Eq. 18) are given in Appendix A. The time-adaptive algorithms have been applied as described by the authors: Ross (2003) for the time stepping based on the saturation changes and Kavetski et al. (2001) for the time stepping based on the truncation errors.

For the standard iterative algorithm, we defined two types of errors to check the convergence: the error based on the maximum change in the state variables between two iterations defined by $\varepsilon_\psi = \max_i \left| \psi_i^{n+1,k+1} - \psi_i^{n+1,k} \right|$ and the truncation error ε_t defined by Eq. (24). Convergence is assumed to be achieved when

$$\varepsilon_\psi < \tau_{\psi,a} + \tau_{\psi,r} \left| \psi_{i\max}^{n+1,k+1} \right|, \tag{27}$$

where $\tau_{\psi,a}$ and $\tau_{\psi,r}$ are the absolute and relative user-defined tolerances and $\psi_{i\max}^{n+1,k+1}$ is the pressure corresponding to ε_ψ

Table 1. Different options of the tested algorithms. Reference to the corresponding equation in parentheses.

	Standard iterative algorithm					Time-adaptive algorithm	
	Time-stepping			Stopping criterion			
	Heuristic (19)	Truncation (25) (26)	Saturation (20) (21)	Pressure (27)	Truncation (28)	Truncation (25) (26)	Saturation (20) (21)
SH_$\Delta\psi$	x			x			
SH_$\Delta\psi$_Δt	x			x	x		
ST_$\Delta\psi$		x		x			
SS_$\Delta\psi$_Δt			x	x	x		
TA_T						x	
TA_S							x

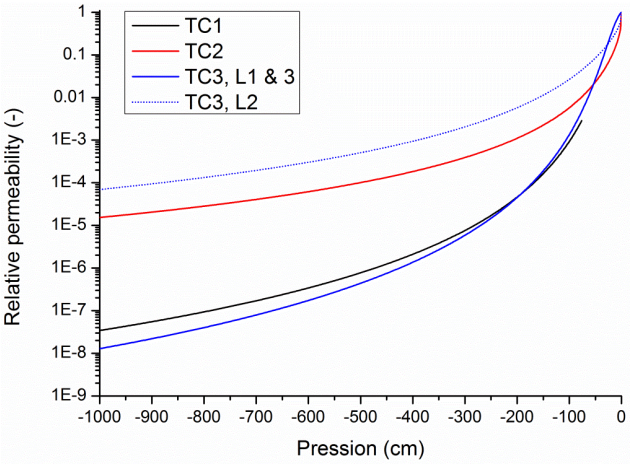


Figure 1. Relative permeability as a function of the pressure for the three test cases (L1, L2 and L3 are the three layers for test case 3).

and when

$$\varepsilon_t < \tau_{t,a} + \tau_{t,r} \left| \psi_{i\max}^{n+1,k+1} \right|, \tag{28}$$

where $\tau_{t,a}$ and $\tau_{t,r}$ have the same meaning as those for the previous criterion but $\psi_{i\max}^{n+1,k+1}$ represents the pressure value corresponding to ε_t.

The tested algorithms are summarized in Table 1. Computations of all possible combinations for the standard iterative scheme have been performed. We present only the four most efficient algorithms. We also analyzed convergence based on the nonlinear residual. It was found to be less restrictive than the previous criteria. Due to the definition of the NR method, the residual tends to zero, but it does not ensure a small value of ε_ψ. Therefore, the results related to the reduction of the nonlinear residuals are not reported.

We investigated three one-dimensional problems with various initial and boundary conditions and hydraulic functions to assess the accuracy, efficiency and computational costs of the different algorithms. The selected test cases represent a range of difficult infiltration problems widely analyzed in the literature.

- TC1: infiltration in a homogeneous initially dry soil with constant prescribed pressure at the surface and prescribed pressure at the bottom (Celia et al., 1990);
- TC2: infiltration in a homogeneous soil initially at hydrostatic equilibrium with a prescribed constant flux at the soil surface and prescribed pressure at the bottom (Miller et al., 1998); and
- TC3: infiltration/evaporation in an initially dry heterogeneous soil, with variable positive and negative fluxes at the surface and free drainage at the base of the soil column (Lehmann and Ackerer, 1998).

For the three test cases, the soil hydraulic functions were described by Mualem–Van Genuchten models (Mualem, 1976; van Genuchten, 1980); see Eqs. (4) and (5).

The required parameters, boundary conditions and initial conditions are summarized in Table 2. The evolution of the relative hydraulic conductivity, the water saturation and the specific moisture capacity with respect to the pressure values are shown in Figs. 1, 2 and 3, respectively. For TC1, the pressure will vary from -1000 to -75 cm only due to the specific conditions of this test case. Therefore, the parameter variations are smaller than those for the other test cases. Since the parameters' variations are more abrupt for test cases 2 and 3, their solutions are more challenging.

Preliminary tests were performed to define the optimal spatial discretization; i.e., a finer spatial discretization provided very similar results for a given convergence criterion and a given time-stepping strategy. Therefore, we can assume that the errors only originate from the time-step size and the linearization.

The following criteria were used for the time-stepping strategy:

- $k_1 = 0.80$, $k_2 = 1.20$, $m_1 = 5$, and $m_2 = 10$, which are the usual values for the heuristic strategy defined by Eq. (19); and

Table 2. Domain size (L), initial conditions (IC), boundary conditions at the soil surface (BC_u) and at the soil bottom (BC_l), saturated hydraulic conductivity (K_s), residual and saturated water contents (θ_r, θ_s) and shape parameters (α, η) for the different test cases. $K_M(t)$ is the hydraulic conductivity of the last grid cell. Length and time units are centimeters and seconds, respectively.

	L	IC	BC_u	BC_l	K_s	θ_r	θ_s	α	η
TC1	30	-1000.0	$\psi = -75$	$\psi = -1000$	9.22×10^{-3}	0.102	0.368	0.0335	2.0
TC2	200	$z - 200$	$q = 3.7 \times 10^{-5}$	$\psi = 0$	7.18×10^{-5}	0.095	0.410	0.019	1.31
TC3	60	-100.0	$q(t)$	$q(t) = K_M(t)$	6.26×10^{-3}	0.0286	0.366	0.028	2.239
	60	-100.0			1.51×10^{-4}	0.106	0.469	0.0104	1.395
	60	-100.0			6.26×10^{-3}	0.0286	0.366	0.028	2.239

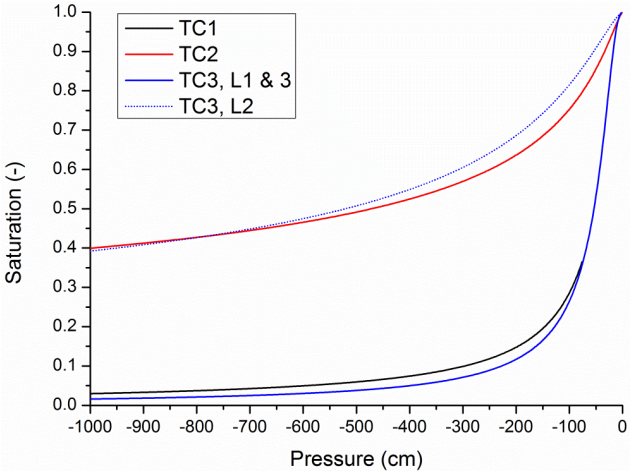


Figure 2. Water saturation as a function of the pressure for the three test cases (L1, L2 and L3 are the three layers for test case 3).

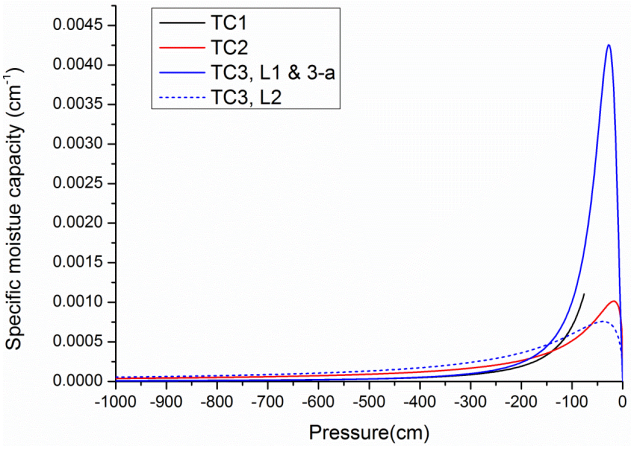


Figure 3. Specific moisture capacity as a function of the pressure for the three test cases (L1, L2 and L3 are the three layers for test case 3).

– $r_{min} = 0.10$, $r_{max} = 4.0$, $s = 0.9$, and $EPS = 10^{-10}$, which are the standard values for the time-stepping scheme based on the time discretization error defined by Eq. (26) (Kavetski et al., 2001).

To perform a consistent comparison of the time-stepping strategies, the maximum allowed change in saturation (see Eqs. 20 and 21) has been evaluated using the maximum change in the pressure, according to the following relationship:

$$\Delta S_{max} \approx \frac{1}{(\theta_{s,imax} - \theta_{r,imax})} \left.\frac{d\theta}{d\psi}\right|_{imax}^{n} \left(\tau_a + \tau_r \left|\psi_{imax}^{n+1,k+1}\right|\right). \quad (29)$$

The simulations have been performed using different values of τ_r and with $\tau_a = 0.0$.

We used several criteria to evaluate the performance of these codes. A typical error used in solving RE is the global cumulative mass balance error defined by

$$MB(t^{n+1}) = \frac{\sum_{i=1}^{M} \Delta z_i \left(\theta_i^{n+1} - \theta_i^0\right)}{\sum_{k=1}^{n+1} \left(q_{in}^k - q_{out}^k\right) \Delta t^k}, \quad (30)$$

where Δz_i is the size of the cell/element i, θ_i^{n+1} is its water content at time t^{n+1}, θ_i^0 is the initial water content, and q_{in}^k and q_{out}^k are the inflow and outflow, respectively, at the domain boundaries at time t^k. M is the number of cells/elements. The fluxes at the boundaries are defined by $q^k = \frac{1}{2}\left(q^k + q^{k-1}\right)$. The mass balance errors were checked for each run but were found to be negligible since we solved the mass-conserving RE form.

While it is necessary to satisfy the global mass balance for an accurate numerical scheme, a low mass balance error is not sufficient to ensure the accuracy of the solution. Therefore, solutions have also been compared with the reference solution obtained using a very fine temporal discretization and the iterative Newton–Raphson method. This comparison

is based on the average relative error defined by

$$\varepsilon_k = \left[\frac{1}{M} \sum_i \frac{\left| \psi_i^{\mathrm{ref}} - \hat{\psi}_i \right|^k}{\left| \psi_i^{\mathrm{ref}} \right|^k} \right]^{1/k}, \quad (31)$$

where M is the number of cells, $\boldsymbol{\psi}^{\mathrm{ref}}$ is the reference solution and $\hat{\boldsymbol{\psi}}$ is the tested numerical solution. ε_1 represents the average absolute relative error (called L_1-norm in the following), ε_2 is the average quadratic error (L_2-norm) and ε_∞ is the highest local relative difference between the numerical and reference solutions (L_∞-norm).

Since the time-adaptive algorithm does not require the computation of the parameters and their derivatives during the iterative procedure, we use N_{sol} to denote the number of times where the system of equations is solved and N_{param} to denote the number of times where the parameters are computed. Of course, these counters are equal to each other for the standard algorithm, which leads to computational costs depending on $2N_{\mathrm{sol}}$. N_{param} is less than N_{sol} for the time-adaptive algorithm. For comparison purposes, the computational costs are estimated by N_{sol} for the standard algorithm and by $(N_{\mathrm{sol}}+N_{\mathrm{param}})/2$ for the time-adaptive algorithm. The efficiency of the algorithms has been evaluated by comparing the computational costs for a given relative tolerance τ_{r}. The errors are presented in the tables and the figures. The figures show some additional results not listed in the tables that already contain much information.

5.1 TC1: Infiltration in a homogenous soil with constant boundary conditions

This test case simulates an infiltration into a homogeneous porous medium. This problem is addressed here because it has been widely analyzed previously by many authors like Bouchemella et al. (2015), Celia et al. (1990), El Kadi and Ling (1993), Rathfelder and Abriola (1994), and Tocci et al. (1997), among others. The computations were performed with a spatial discretization of 0.1 cm. The initial time-step size was set to 1.0×10^{-5} s, and the maximum time-step size was set to 400 s.

The results for the iterative and time-adaptive algorithms are presented in Tables 3 and 4, respectively. When both convergence criteria are used (algorithms SH_$\Delta\psi$_Δt and SS_$\Delta\psi$_Δt), N_{trunc} represents the number of times where the truncation error is the most restrictive condition. For the heuristic time-stepping schemes, the convergence is mostly linked to the truncation error (N_{trunc} is close to N_{sol}), whereas when the saturation time-stepping scheme is used, the most restrictive criterion is the maximum difference in the pressure.

When the time-stepping scheme is based on saturation, for both iterative and time-adaptive algorithms, the number of iterations required to solve the problem is proportional to the relative tolerance. Therefore, highly accurate solutions incur high computational costs.

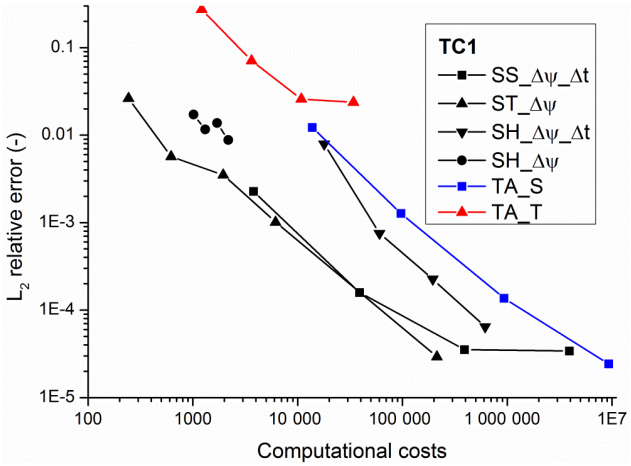


Figure 4. Evolution of the L_2 relative error with computational costs for TC1.

For the time-adaptive scheme, the number of parameter changes N_{param} is close to the number of iterations for low tolerance values. Small tolerance values lead to small time steps, avoiding time-step adjustments. This is not the case for larger tolerance values that lead to larger time steps and therefore to additional iterations (see for example TA_T for the tolerance of $\tau_{\mathrm{r}} = 10^{-2}$ – Table 4).

The three types of errors provide the same information. The best solution for one type of error is also the best solution for the other two.

On average, the iterative algorithm is faster than the time-adaptive algorithm that requires more iterations for a given error. This is also shown in Fig. 4 that presents the convergence rate of the L_2-norm with respect to the computational costs, i.e., the number of iterations or number of iterations and number of parameter changes. The time-adaptive algorithm with time stepping based on the truncation errors performs quite poorly compared to the other algorithms. Irrespective of the tolerance, this algorithm leads to a wetting front moving faster (Fig. 5).

When the relative tolerance is set to a very low value ($\tau_{\mathrm{r}} = 10^{-5}$), the iterative scheme with time stepping based on the saturation changes shows behavior that is different from that found for the less restrictive tolerance. The criterion based on truncation errors is no longer significant ($N_{\mathrm{trunc}} = 252$), possibly explaining why the accuracy of the scheme remains constant. This also indicates that errors due to time discretization have to be handled, either in the convergence criterion or in the time-stepping strategy.

For this test case, the most efficient algorithms are the iterative algorithms using the time-stepping strategy based on truncation error (ST_$\Delta\psi$) or based on the saturation changes (SS_$\Delta\psi$_Δt). Saturation-based time-stepping strategies (SS_$\Delta\psi$_Δt and TA_S) show a linear decrease

Table 3. Relative errors and number of iterations obtained for the iterative algorithm depending on different convergence criteria for TC1.

Tol.	Algorithm	L_1	L_2	L_∞	N_{trunc}	N_{sol}
10^{-5}	SH_$\Delta\psi$	1.918×10^{-3}	8.829×10^{-3}	0.106		2177
	SH_$\Delta\psi$_Δt	8.391×10^{-6}	6.459×10^{-5}	8.782×10^{-4}	542 371	615 880
	ST_$\Delta\psi$	3.968×10^{-4}	1.045×10^{-3}	3.512×10^{-3}		6160
	SS_$\Delta\psi$_Δt	1.136×10^{-5}	3.406×10^{-5}	2.817×10^{-4}	252	392 0446
10^{-4}	SH_$\Delta\psi$	2.557×10^{-3}	1.375×10^{-2}	0.168		1701
	SH_$\Delta\psi$_Δt	7.818×10^{-5}	2.259×10^{-4}	1.593×10^{-3}	170 438	194 420
	ST_$\Delta\psi$	1.331×10^{-3}	1.316×10^{-3}	1.181×10^{-2}		1950
	SS_$\Delta\psi$_Δt	8.607×10^{-6}	3.525×10^{-5}	3.899×10^{-4}	154 597	392 041
10^{-3}	SH_$\Delta\psi$	3.956×10^{-3}	1.166×10^{-2}	0.125		1312
	SH_$\Delta\psi$_Δt	2.320×10^{-4}	7.553×10^{-4}	7.883×10^{-3}	52 723	60 303
	ST_$\Delta\psi$	2.241×10^{-3}	5.702×10^{-3}	1.792×10^{-2}		620
	SS_$\Delta\psi$_Δt	6.567×10^{-5}	1.585×10^{-4}	1.453×10^{-3}	9895	39 110
10^{-2}	SH_$\Delta\psi$	6.559×10^{-3}	1.716×10^{-2}	0.119		1018
	SH_$\Delta\psi$_Δt	2.224×10^{-3}	7.923×10^{-3}	7.111×10^{-2}	15 540	17 888
	ST_$\Delta\psi$	9.954×10^{-3}	2.630×10^{-2}	8.727×10^{-2}		243
	SS_$\Delta\psi$_Δt	8.283×10^{-4}	2.271×10^{-3}	1.478×10^{-2}	862	3804

Table 4. Relative errors and number of iterations obtained for the time-adaptive algorithm depending on different convergence criteria for TC1.

Tol.	Algorithm	L_1	L_2	L_∞	N_{param}	N_{sol}
10^{-5}	TA_T	5.016×10^{-3}	2.376×10^{-2}	0.269	32 197	35 938
	TA_S	6.152×10^{-6}	2.429×10^{-5}	2.561×10^{-4}	9 316 700	9 322 946
10^{-4}	TA_T	5.598×10^{-3}	2.580×10^{-2}	0.284	10 169	11 520
	TA_S	2.839×10^{-5}	1.363×10^{-4}	1.654×10^{-3}	931 616	938 144
10^{-3}	TA_T	1.524×10^{-2}	7.085×10^{-2}	0.822	3231	4032
	TA_S	2.537×10^{-4}	1.271×10^{-3}	1.568×10^{-2}	93 114	100 898
10^{-2}	TA_T	6.241×10^{-2}	0.274	2.459	1023	1402
	TA_S	2.519×10^{-3}	1.224×10^{-2}	0.142	9267	18 292

in L_2 with computational costs. For very high precision ($L_2 < 10^{-4}$), ST_$\Delta\psi$ outperforms the other algorithms. No convincing explanation has been found for the insignificant change in accuracy for SS_$\Delta\psi$_Δt at high precision.

5.2 TC2: Infiltration in a homogenous soil with hydrostatic initial conditions

This test case models an infiltration in a 200 cm vertical column of unconsolidated clay loam with non-uniform grain size distribution and was considered by Miller et al. (1998) to be a very challenging test. This problem was found to be more challenging from the numerical point of view compared to TC1 due to the relative permeability function that enhances the nonlinear behavior of Richards' equation (Figs. 1, 2, and 3). The cell size has been set to 0.125 cm, the initial time step to 10^{-5} s and the maximum time-step magnitude to 1000 s.

The different norms for the iterative and time-adaptive schemes are given in Tables 5 and 6.

Investigation of this test case leads to similar qualitative conclusions when the time-stepping scheme is based on the saturation differences (SS_$\Delta\psi$_Δt and TA_S). The standard scheme SH_$\Delta\psi$ fails to provide an accurate solution within a reasonable number of iterations (less than 10^7).

The most efficient methods are the schemes using the time-stepping strategy based on truncation errors (Fig. 6). However, as found for TC1, the adaptive time algorithm TA_T failed to provide highly accurate results (L_2-norm error less than approximately 4.5×10^{-4}).

Figure 7 shows the time-step magnitudes for approximately equal L_2-norms for the two time-adaptive algorithms and for the iterative algorithm using truncation errors

Table 5. Relative errors and number of iterations obtained for the iterative algorithm depending on different convergence criteria for TC2 (n.c.: non convergence in less than 10^7 iterations).

Tol.	Algorithm	L_1	L_2	L_∞	N_{trunc}	N_{sol}
10^{-5}	SH_$\Delta\psi$	6.966×10^{-3}	1.818×10^{-2}	5.878×10^{-2}		573
	SH_$\Delta\psi$_Δt	$3.697. \times 10^{-4}$	9.766×10^{-4}	3.332×10^{-3}	53 769	59 643
	ST_$\Delta\psi$	1.578×10^{-4}	4.254×10^{-4}	2.451×10^{-3}		3503
	SS_$\Delta\psi$_Δt	–	–	–	–	n.c.
10^{-4}	SH_$\Delta\psi$	6.966×10^{-3}	1.818×10^{-2}	5.878×10^{-2}		509
	SH_$\Delta\psi$_Δt	6.968×10^{-4}	1.979×10^{-3}	5.726×10^{-3}	16 557	18 428
	ST_$\Delta\psi$	5.814×10^{-4}	1.492×10^{-3}	6.711×10^{-3}		1033
	SS_$\Delta\psi$_Δt	3.279×10^{-6}	1.239×10^{-5}	8.603×10^{-5}	0	2 474 120
10^{-3}	SH_$\Delta\psi$	6.966×10^{-3}	1.818×10^{-2}	5.878×10^{-2}		410
	SH_$\Delta\psi$_Δt	3.699×10^{-3}	9.761×10^{-3}	3.275×10^{-2}	4830	5444
	ST_$\Delta\psi$	1.553×10^{-3}	4.226×10^{-3}	2.457×10^{-2}		317
	SS_$\Delta\psi$_Δt	2.355×10^{-5}	6.230×10^{-5}	2.341×10^{-4}	0	247 426
10^{-2}	SH_$\Delta\psi$	6.892×10^{-3}	1.800×10^{-2}	5.780×10^{-2}		309
	SH_$\Delta\psi$_Δt	9.135×10^{-3}	2.409×10^{-2}	7.925×10^{-2}	376	580
	ST_$\Delta\psi$	2.756×10^{-3}	1.134×10^{-2}	7.715×10^{-2}		180
	SS_$\Delta\psi$_Δt	2.973×10^{-4}	7.884×10^{-4}	3.252×10^{-3}	0	24 757

Table 6. Relative errors and number of iterations obtained for the time-adaptive algorithm depending on different convergence criteria for TC2.

Tol.	Algorithm	L_1	L_2	L_∞	N_{param}	N_{sol}
10^{-5}	TA_T	1.230×10^{-4}	4.563×10^{-4}	3.346×10^{-3}	3089	3098
	TA_S	8.741×10^{-6}	2.308×10^{-5}	7.905×10^{-5}	1 136 193	1 136 199
10^{-4}	TA_T	1.572×10^{-3}	4.497×10^{-3}	2.404×10^{-2}	986	987
	TA_S	2.701×10^{-5}	7.219×10^{-5}	3.095×10^{-4}	113 616	113 616
10^{-3}	TA_T	4.707×10^{-3}	1.346×10^{-2}	7.169×10^{-2}	323	323
	TA_S	1.754×10^{-4}	4.844×10^{-4}	2.391×10^{-3}	11 358	11 358
10^{-2}	TA_T	5.220×10^{-3}	1.683×10^{-2}	0.101	135	135
	TA_S	1.596×10^{-3}	4.444×10^{-3}	2.243×10^{-2}	1132	1132

for time stepping (4.254×10^{-4} within 3503 iterations for ST_$\Delta\psi$, 4.563×10^{-4} within 3098 iterations for TA_T and 4.844×10^{-4} within 11358 iterations for TA_S). The time-step evolution is very similar for the three strategies: a linear increase until around 0.1 s, followed by a very slow increase until 20–30 s and a regular increase until the end of the simulation. ST_$\Delta\psi$ and TA_T strategies lead to the same time steps when the time reaches 1 s. The time-step sizes remain smaller for TA_S, which explains the significantly higher number of iterations required to solve this test case.

5.3 TC3: Infiltration/evaporation in a heterogeneous soil

This case study simulates infiltration in an initially dry heterogeneous soil with a succession of rainfall and evaporations as upper boundary conditions during 35 days. This problem differs from the two previous cases by the soil heterogeneity and also by the non-monotonic boundary conditions at the soil surface. It is expected that non-monotonic discontinuous boundary conditions will increase the difficulty in finding accurate solutions. The soil profile consists of three 60 cm thick layers. The layers are discretized using cells with the size of 0.10 cm. The prescribed fluxes are changing every day. For a given time, these fluxes are linearly interpolated. To avoid an overly rough time discretization of these boundary conditions, the maximum time-step magnitude has been fixed at 0.20 days. The initial time step is set to 10^{-5} days.

The relative errors estimated by the iterative algorithms and the time-adaptive algorithms are presented in Tables 7 and 8, respectively, and are plotted in Fig. 8.

Table 7. Relative errors and number of iterations obtained for the iterative algorithm depending on different convergence criteria for TC3 (n.c.: non convergence in less than 10^7 iterations, * convergence failed for 10^{-3}, $\tau_r = 0.90 \times 10^{-3}$).

Tol.	Algorithm	L_1	L_2	L_∞	N_{trunc}	N_{sol}
10^{-5}	SH_$\Delta\psi$	9.994×10^{-3}	1.119×10^{-2}	1.554×10^{-2}		1644
	SH_$\Delta\psi$_Δt	6.612×10^{-4}	7.346×10^{-4}	1.116×10^{-3}	17 1636	190 588
	ST_$\Delta\psi$	6.830×10^{-4}	7.775×10^{-4}	1.648×10^{-3}		16 984
	SS_$\Delta\psi$_Δt	7.185×10^{-5}	7.935×10^{-5}	1.297×10^{-4}	197 481	164 6346
10^{-4}	SH_$\Delta\psi$	6.664×10^{-3}	7.280×10^{-3}	1.033×10^{-2}		1734
	SH_$\Delta\psi$_Δt	3.512×10^{-3}	3.898×10^{-3}	5.811×10^{-3}	57 312	63 956
	ST_$\Delta\psi$	1.300×10^{-3}	1.517×10^{-3}	2.412×10^{-3}		6504
	SS_$\Delta\psi$_Δt	5.380×10^{-5}	6.536×10^{-5}	1.010×10^{-4}	41 073	186 351
10^{-3}	SH_$\Delta\psi$	–	–	–		n.c.
	SH_$\Delta\psi$_Δt	2.625×10^{-3}	2.899×10^{-3}	4.971×10^{-3}	22 047	24 779
	ST_$\Delta\psi$	4.730×10^{-3}	5.422×10^{-3}	1.036×10^{-2}		1297*
	SS_$\Delta\psi$_Δt	7.569×10^{-4}	8.820×10^{-4}	1.402×10^{-3}	16 474	31 276
10^{-2}	SH_$\Delta\psi$	–	–	–		n.c.
	SH_$\Delta\psi$_Δt	5.493×10^{-3}	6.306×10^{-3}	1.171×10^{-3}	7438	8812
	ST_$\Delta\psi$	6.621×10^{-3}	7.402×10^{-3}	1.042×10^{-2}		810
	SS_$\Delta\psi$_Δt	7.511×10^{-3}	8.780×10^{-3}	1.378×10^{-2}	5838	7535

Table 8. Relative errors and number of iterations obtained for the time-adaptive algorithm depending on different convergence criteria for TC3.

Tol.	Algorithm	L_1	L_2	L_∞	N_{param}	N_{sol}
10^{-5}	TA_T	9.814×10^{-3}	9.949×10^{-3}	1.286×10^{-2}	8369	8703
	TA_S	7.980×10^{-5}	8.797×10^{-5}	1.472×10^{-4}	1 357 075	1 357 160
10^{-4}	TA_T	1.731×10^{-2}	1.760×10^{-2}	2.748×10^{-2}	2653	2934
	TA_S	1.067×10^{-4}	1.247×10^{-4}	1.997×10^{-4}	135 386	135 498
10^{-3}	TA_T	2.922×10^{-2}	3.105×10^{-2}	4.545×10^{-2}	889	1153
	TA_S	1.433×10^{-4}	1.788×10^{-4}	3.367×10^{-4}	13 314	13 397
10^{-2}	TA_T	1.996×10^{-2}	2.449×10^{-2}	5.536×10^{-2}	347	515
	TA_S	1.851×10^{-3}	2.051×10^{-3}	3.925×10^{-3}	1232	1283

The standard iterative scheme fails to converge within the maximum number of iterations (10^7) when the tolerance is not sufficiently restrictive. The detailed analyses of the computation showed that the time-step size was quite large compared to the more restrictive conditions until day 28.0, where the infiltration fluxes were equal to 1.50 cm day^{-1} and where the conditions were near saturation due to the previous infiltration period. This led to a decrease in the time step to close to the minimum value (10^{-8} s), causing the procedure to stop. More restrictive conditions lead to smaller time steps from the beginning of the simulation and a better approximation of the solutions during the entire simulation.

The iterative scheme coupled with the truncation-based time-stepping strategy showed surprisingly unstable behavior for $\tau_r = 10^{-3}$. The scheme did not converge for $\tau_r \in [0.96 \times 10^{-3}; 1.04 \times 10^{-3}]$. The results presented in Table 7 and Fig. 8 are obtained for $\tau_r = 0.90 \times 10^{-3}$. At this stage of our work, we were not able to provide a meaningful explanation for this effect.

The time-adaptive algorithm with the saturation-based time-stepping scheme is the most efficient for an L_2-norm greater than 10^{-4}. For more accurate results, the iterative method with the time-stepping strategy using the truncation error must be preferred. The impact of the time-stepping strategy for these two algorithms is shown in Fig. 9 for approximately the same L_2-norm (2.051×10^{-3} within 1283 iterations for TA_S and 1.517×10^{-3} within 6504 iterations for ST_$\Delta\psi$). The time-step changes are related to the boundary condition variations, as expected. The strategy based on the saturation variation leads to a longer time step than the strategy using the time truncation error. This difference can be quite important (see the simulation between days 25

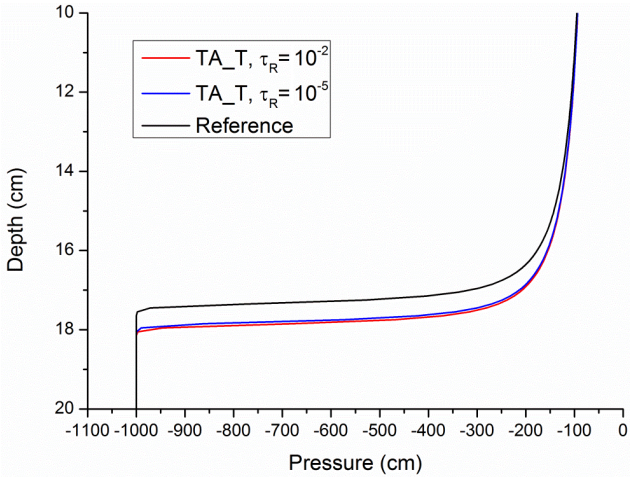


Figure 5. Pressure profiles in the domain for the TA_T algorithm.

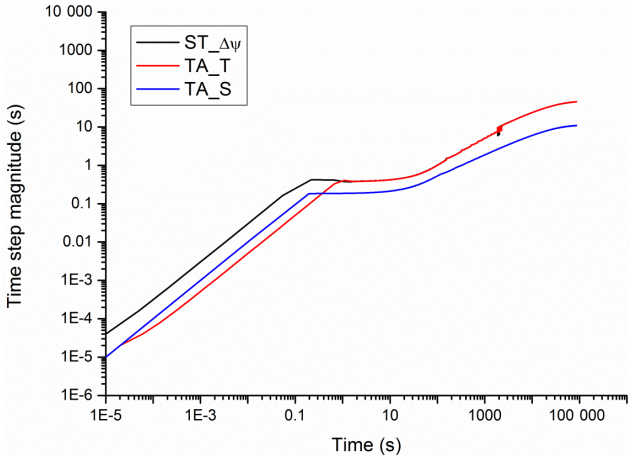


Figure 7. Time-step magnitudes during the simulation for TC2.

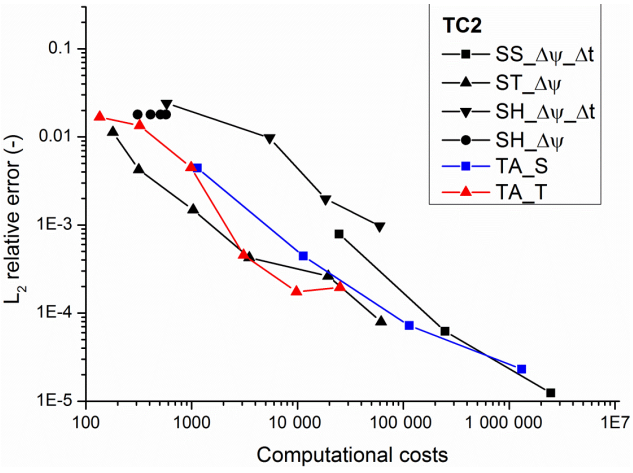


Figure 6. Evolution of the L_2 relative error with computational costs for TC2.

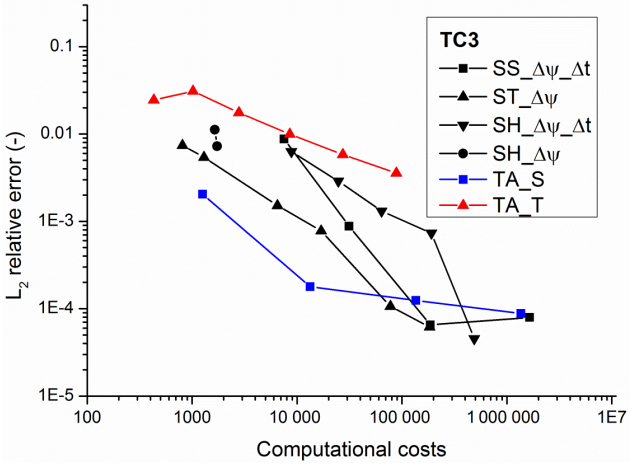


Figure 8. Evolution of the L_2 relative error with computational costs for TC3.

and 30). The consequences of this difference are a reduced number of iterations but also a less accurate computation, irrespective of the error norm.

6 Summary and conclusions

The solution of RE is complex and very time-consuming due to its highly nonlinear properties. Several algorithms have been tested for the mixed form of the Richards equation, including time-adaptive methods. Based on the numerical examples that differ in their parameters (level of nonlinearity) and in their initial and boundary conditions, the conclusions and recommendations are the following.

1. Our numerical developments showed that the method suggested by Ross (2003) in its implicit formulation can be considered as a Newton–Raphson method with a time-adaptive algorithm.

2. The different algorithms have different convergence rates (accuracy improvement of the scheme as a function of the computational costs). Therefore, an algorithm can be very efficient for a given accuracy and less efficient for another level of precision. However, for these three test cases and, on average, the best performance in terms of efficiency was obtained using a stopping criterion based on truncation error with its corresponding time-step strategy (ST_$\Delta\psi$). Similar results were obtained by Kavetski et al. (2001) for the pressure-based RE and by Kavetski and Binning (2004) for the moisture-based RE.

3. The mass balance is not a good criterion for the evaluation of the results because the mixed form preserves the mass balance, irrespective of the pressure distribution within the profile.

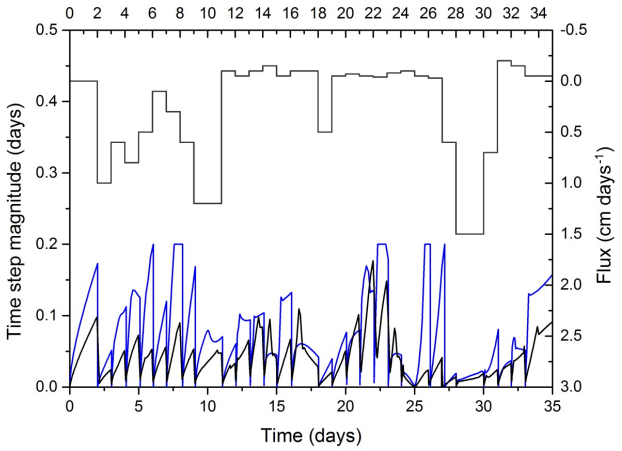


Figure 9. Time-step magnitudes during the simulation for TC3 for the time-stepping strategy based on truncation error (TA_S in blue, TA_T in black, time-varying boundary conditions at the top).

4. The time truncation error should be implemented in numerical codes using the standard iterative procedure. The use of the maximum variable difference between two successive iterations only, which is usually implemented, does not provide any information about the accuracy of the time derivative approximation.

Our one-dimensional examples showed that the time-adaptive algorithm TA_T is very sensitive to the type of problem to be solved. The time-adaptive algorithm TA_S was less efficient than the usual schemes. However, for a larger number of elements like in two-dimensional or three-dimensional problems, this conclusion might be different because the time dedicated to the computation of the parameters can be significantly higher, unless tabulated values are used to evaluate the parameters and the required derivatives.

Depending on the type of the problem that must be solved (parameter behavior with respect to the pressure, time variations of the boundary conditions), the time truncation errors may be predominant compared to the error corresponding to the pressure changes between two successive iterations. Therefore, we recommend the implementation of this stopping criteria associated with the time-stepping strategy as defined by Kavetski et al. (2001).

Data availability. We did not use data. The computer program is available upon request.

Appendix A

The numerical method used in the paper is implicit standard finite difference. For a cell i of the grid, the unsaturated flow Eq. (4) can be discretized in the following way:

$$\begin{cases} \dfrac{\theta_i^{n+1}-\theta_i^n}{\Delta t}+S_{\mathrm{w}}s_0\dfrac{\psi_i^{n+1}-\psi_i^n}{\Delta t}+\dfrac{q_{i+}^{n+1}-q_{i-}^{n+1}}{\Delta z_i}=f_i,\\ q_{i-}^{n+1}=-K_{i-}\left(\dfrac{\psi_i^{n+1}-\psi_{i-1}^{n+1}}{\Delta z_{i-}}-1\right),\\ q_{i+}^{n+1}=-K_{i+}\left(\dfrac{\psi_{i+1}^{n+1}-\psi_i^{n+1}}{\Delta z_{i+}}-1\right), \end{cases} \tag{A1}$$

where n is the time step, K_{i-} is the inter-block conductivity between cell i and $(i-1)$ defined by $K_{i-}=\frac{\Delta z_{i-1}K(\psi_{i-1})+\Delta z_i K(\psi_i)}{\Delta z_{i-1}+\Delta z_i}$, and K_{i+} is the inter-block conductivity between cell i and $(i+1)$ defined by $K_{i+}=\frac{\Delta z_i K(\psi_i)+\Delta z_{i+1}K(\psi_{i+1})}{\Delta z_i+\Delta z_{i+1}}$. $\Delta z_{i-}=\frac{1}{2}(\Delta z_{i-1}+\Delta z_i)$ is the distance between the center of cell $(i-1)$ and i. $\Delta z_{i+}=\frac{1}{2}(\Delta z_i+\Delta z_{i+1})$ is the distance between the center of cell i and $(i+1)$.

The residual is

$$\begin{aligned} R(\psi_i^{n+1,k})=&\Delta z_i\left(\theta_i^{n+1,k}-\theta_i^n\right)+\Delta z_i S_{\mathrm{w}}s_0\left(\psi_i^{n+1,k}-\psi_i^n\right)\\ &+\Delta t\left(q_{i+}^{n+1,k}-q_{i-}^{n+1,k}\right)-\Delta t\,\Delta z_i f_i \end{aligned} \tag{A2}$$

where k is the iteration counter.

The residual derivatives are

$$\begin{aligned} \frac{\partial R(\psi_i^{n+1,k})}{\partial\psi_{i-1}^{n+1,k}}=&-\Delta t\frac{\partial q_{i-}^{n+1,k}}{\partial\psi_{i-1}^{n+1,k}},\\ \frac{\partial R(\psi_i^{n+1,k})}{\partial\psi_i^{n+1,k}}=&\Delta z_i\frac{d\theta_i^{n+1,k}}{d\psi_i^{n+1,k}}+\Delta z_i S_{\mathrm{w}}s_0\\ &+\Delta t\left(\frac{\partial q_{i+}^{n+1,k}}{\partial\psi_i^{n+1,k}}-\frac{\partial q_{i-}^{n+1,k}}{\partial\psi_i^{n+1,k}}\right),\\ \frac{\partial R(\psi_i^{n+1,k})}{\partial\psi_{i+1}^{n+1,k}}=&\Delta t\frac{\partial q_{i+}^{n+1,k}}{\partial\psi_{i+1}^{n+1,k}}. \end{aligned} \tag{A3}$$

Therefore, the system to be solved is

$$\begin{aligned} &-\Delta t\frac{\partial q_{i-}^{n+1,k}}{\partial\psi_{i-1}^{n+1,k}}\Delta\psi_{i-1}^{n+1,k+1}\\ &+\left[\Delta z_i\frac{d\theta_i^{n+1,k}}{d\psi_i^{n+1,k}}+\Delta z_i S_{\mathrm{w}}s_0+\Delta t\left(\frac{\partial q_{i+}^{n+1,k}}{\partial\psi_i^{n+1,k}}-\frac{\partial q_{i-}^{n+1,k}}{\partial\psi_i^{n+1,k}}\right)\right]\\ &\Delta\psi_i^{n+1,k+1}+\Delta t\frac{\partial q_{i+}^{n+1}}{\partial\psi_{i+1}^{n+1,k}}\Delta\psi_{i+1}^{n+1,k+1}\\ &=-\Delta z_i\left(\theta_i^{n+1,k}-\theta_i^n\right)-\Delta z_i S_{\mathrm{w}}s_0\left(\psi_i^{n+1,k}-\psi_i^n\right)\\ &-\Delta t\left(q_{i+}^{n+1,k}-q_{i-}^{n+1,k}\right)+\Delta t\,\Delta z_i f_i, \end{aligned} \tag{A4}$$

with the following derivatives of the fluxes $q_{i-}^{n+1,k}$,

$$\begin{cases} \dfrac{\partial q_{i-}^{n+1,k}}{\partial\psi_{i-1}^{n+1,k}}=-\dfrac{\partial K_{i-}^{n+1,k}}{\partial\psi_{i-1}^{n+1,k}}\left(\dfrac{\psi_i^{n+1,k}-\psi_{i-1}^{n+1,k}}{\Delta z_{i-}}-1\right)+\dfrac{K_{i-}^{n+1,k}}{\Delta z_{i-}},\\ \dfrac{\partial q_{i-}^{n+1,k}}{\partial\psi_i^{n+1,k}}=-\dfrac{\partial K_{i-}^{n+1,k}}{\partial\psi_i^{n+1,k}}\left(\dfrac{\psi_i^{n+1,k}-\psi_{i-1}^{n+1,k}}{\Delta z_{i-}}-1\right)-\dfrac{K_{i-}^{n+1,k}}{\Delta z_{i-}}, \end{cases} \tag{A5}$$

and $q_{i+}^{n+1,k}$:

$$\begin{cases} \dfrac{\partial q_{i+}^{n+1,k}}{\partial\psi_i^{n+1,k}}=-\dfrac{\partial K_{i+}^{n+1,k}}{\partial\psi_i^{n+1,k}}\left(\dfrac{\psi_{i+1}^{n+1,k}-\psi_i^{n+1,k}}{\Delta z_{i+}}-1\right)+\dfrac{K_{i+}^{n+1,k}}{\Delta z_{i+}},\\ \dfrac{\partial q_{i+}^{n+1,k}}{\partial\psi_{i+1}^{n+1,k}}=-\dfrac{\partial K_{i+}^{n+1,k}}{\partial\psi_{i+1}^{n+1,k}}\left(\dfrac{\psi_{i+1}^{n+1,k}-\psi_i^{n+1,k}}{\Delta z_{i+}}-1\right)-\dfrac{K_{i+}^{n+1,k}}{\Delta z_{i+}}. \end{cases} \tag{A6}$$

The component of the vector of the residuals $\boldsymbol{R}$ is given by Eq. (A2) and the coefficients of the matrix $\boldsymbol{R}'$ for cell i are

$$\begin{aligned} \boldsymbol{R}'_{i-1,i}=&\Delta t\left[\frac{\partial K_{i-}^{n+1,k}}{\partial\psi_{i-1}^{n+1,k}}\left(\frac{\psi_i^{n+1,k}-\psi_{i-1}^{n+1,k}}{\Delta z_{i-}}-1\right)-\frac{K_{i-}^{n+1,k}}{\Delta z_{i-}}\right],\\ \boldsymbol{R}'_{i,i}=&\Delta z_i\left[\frac{d\theta_i^{n+1,k}}{\mathrm{d}\psi_i^{n+1,k}}+S_{\mathrm{w}}s_0\right]\\ &-\Delta t\left[\frac{\partial K_{i+}^{n+1,k}}{\partial\psi_i^{n+1,k}}\left(\frac{\psi_{i+1}^{n+1,k}-\psi_i^{n+1,k}}{\Delta z_{i+}}\right)-\frac{K_{i+}^{n+1,k}}{\Delta z_{i+}}\right]\\ &+\Delta t\left[\frac{\partial K_{i-}^{n+1,k}}{\partial\psi_i^{n+1,k}}\left(\frac{\psi_i^{n+1,k}-\psi_{i-1}^{n+1,k}}{\Delta z_{i-}}-1\right)+\frac{K_{i-}^{n+1,k}}{\Delta z_{i-}}\right],\\ \boldsymbol{R}'_{i,i+1}=&-\Delta t\left[\frac{\partial K_{i+}^{n+1,k}}{\partial\psi_{i+1}^{n+1,k}}\left(\frac{\psi_{i+1}^{n+1,k}-\psi_i^{n+1,k}}{\Delta z_{i+}}-1\right)+\frac{K_{i+}^{n+1,k}}{\Delta z_{i+}}\right]. \end{aligned} \tag{A7}$$

In the case of prescribed flux at the upper boundary, the residual is written as

$$\begin{aligned} \boldsymbol{R}_1(\psi_1^{n+1,k})=&\Delta z_1\left[\left(\theta_1^{n+1,k}-\theta_1^n\right)+S_{\mathrm{w}}s_0\left(\psi_1^{n+1,k}-\psi_1^n\right)\right]\\ &+\Delta t\left(q_{1+}^{n+1}-q_{BC}\right)-\Delta t\,\Delta z_1 f_1. \end{aligned} \tag{A8}$$

Using the derivatives as defined in Eqs. (A5) and (A6), the matrix coefficients are changed as follows:

$$\begin{aligned} \boldsymbol{R}'_{1,1}=&\Delta z_1\left(\frac{d\theta_1^{n+1,k}}{d\psi_1^{n+1,k}}+S_{\mathrm{w}}s_0\right)\\ &-\Delta t\left[\frac{\partial K_{1+}^{n+1,k}}{\partial\psi_1^{n+1,k}}\left(\frac{\psi_2^{n+1,k}-\psi_1^{n+1,k}}{\Delta z_{1+}}-1\right)-\frac{K_{1+}^{n+1,k}}{\Delta z_{1+}}\right],\\ \boldsymbol{R}'_{1,2}=&-\Delta t\left[\frac{\partial K_{1+}^{n+1,k}}{\partial\psi_2^{n+1,k}}\left(\frac{\psi_2^{n+1,k}-\psi_1^{n+1,k}}{\Delta z_{1+}}-1\right)+\frac{K_{1+}^{n+1,k}}{\Delta z_{1+}}\right]. \end{aligned} \tag{A9}$$

If the flux is applied at the bottom of the profile, similar developments lead to the residual

$$\boldsymbol{R}_N = \Delta z_N \left[\left(\theta_N^{n+1,k} - \theta_N^n \right) + S_\mathrm{w} s_0 \left(\psi_N^{n+1,k} - \psi_N^n \right) \right] + \Delta t \left(q_{BC} - q_{N-}^{n+1,k} \right) - \Delta t \Delta z_N f_N \quad \text{(A10)}$$

and its derivatives

$$\boldsymbol{R}'_{N-1,N} = \Delta t \left[\frac{\partial K_{N-}^{n+1,k}}{\partial \psi_{N-1}^{n+1,k}} \left(\frac{\psi_N^{n+1,k} - \psi_{N-1}^{n+1,k}}{\Delta z_{N-}} - 1 \right) - \frac{K_{N-}^{n+1,k}}{\Delta z_{N-}} \right],$$

$$\boldsymbol{R}'_{N,N} = \Delta z_N \left(\frac{d\theta_N^{n+1,k}}{d\psi_N^{n+1,k}} + S_\mathrm{w} s_0 \right) + \Delta t \left[\frac{\partial K_{N-}^{n+1,k}}{\partial \psi_N^{n+1,k}} \left(\frac{\psi_N^{n+1,k} - \psi_{N-1}^{n+1,k}}{\Delta z_{N-}} - 1 \right) + \frac{K_{N-}^{n+1,k}}{\Delta z_{N-}} \right]. \quad \text{(A11)}$$

If the pressure is described at the top of the soil, the corresponding flux is defined by

$$q_{1-}^{n+1,k} = -K_{1-} \left(\frac{\psi_1^{n+1,k} - \psi_{BC}}{\Delta z_1/2} - 1 \right), \quad \text{(A12)}$$

and the derivative is

$$\frac{\partial q_{1-}^{n+1,k}}{\partial \psi_1^{n+1,k}} = -\frac{\partial K_{1-}^{n+1,k}}{\partial \psi_1^{n+1,k}} \left(\frac{\psi_1^{n+1,k} - \psi_{BC}}{\Delta z_1/2} - 1 \right) - \frac{K_{1-}^{n+1,k}}{\Delta z_1/2}. \quad \text{(A13)}$$

The corresponding residual and the matrix coefficients are

$$\boldsymbol{R}_1 = \Delta z_1 \left[\left(\theta_1^{n+1,k} - \theta_1^n \right) + S_\mathrm{w} s_0 \left(\psi_1^{n+1,k} - \psi_1^n \right) \right] + \Delta t \left(q_{1+}^{n+1,k} - q_{1-}^{n+1,k} \right) - \Delta t \Delta z_1 f_1 \quad \text{(A14)}$$

and

$$\boldsymbol{R}'_{1,1} = \Delta z_1 \left(\frac{d\theta_1^{n+1,k}}{d\psi_1^{n+1,k}} + S_\mathrm{w} s_0 \right) - \Delta t \left[\frac{\partial K_{1+}^{n+1,k}}{\partial \psi_1^{n+1,k}} \left(\frac{\psi_2^{n+1,k} - \psi_1^{n+1,k}}{\Delta z_{1+}} - 1 \right) - \frac{K_{1+}^{n+1,k}}{\Delta z_{1+}} \right] + \Delta t \left[\frac{\partial K_{1-}^{n+1,k}}{\partial \psi_1^{n+1,k}} \left(\frac{\psi_1^{n+1,k} - \psi_{BC}}{\Delta z_1/2} - 1 \right) + \frac{K_{1-}^{n+1,k}}{\Delta z_1/2} \right]$$

$$\boldsymbol{R}'_{1,2} = -\Delta t \left[\frac{\partial K_{1+}^{n+1,k}}{\partial \psi_2^{n+1,k}} \left(\frac{\psi_2^{n+1,k} - \psi_1^{n+1,k}}{\Delta z_{1+}} - 1 \right) + \frac{K_{1+}^{n+1,k}}{\Delta z_{1+}} \right]. \quad \text{(A15)}$$

Similarly, if the pressure is prescribed at the soil column's bottom, we have

$$\boldsymbol{R}_N = \Delta z_N \left[\left(\theta_N^{n+1,k} - \theta_N^n \right) + S_\mathrm{w} s_0 \left(\psi_N^{n+1,k} - \psi_N^n \right) \right] + \Delta t \left(q_{N+}^{n+1,k} - q_{N-}^{n+1,k} \right) - \Delta t \Delta z_N f_N \quad \text{(A16)}$$

and

$$\boldsymbol{R}'_{N-1,N} = \Delta t \left[\frac{\partial K_{N-}^{n+1,k}}{\partial \psi_{N-1}^{n+1,k}} \left(\frac{\psi_N^{n+1,k} - \psi_{N-1}^{n+1,k}}{\Delta z_{N-}} - 1 \right) - \frac{K_{N-}^{n+1,k}}{\Delta z_{N-}} \right],$$

$$\boldsymbol{R}'_{N,N} = \Delta z_N \left(\frac{d\theta_N^{n+1,k}}{d\psi_N^{n+1,k}} + S_\mathrm{w} s_0 \right) - \Delta t \left[\frac{\partial K_{N+}^{n+1,k}}{\partial \psi_N^{n+1,k}} \left(\frac{\psi_{BC} - \psi_N^{n+1,k}}{\Delta z_N/2} - 1 \right) - \Delta t \frac{K_{N+}^{n+1,k}}{\Delta z_N/2} \right] + \Delta t \left[\frac{\partial K_{N-}^{n+1,k}}{\partial \psi_N^{n+1,k}} \left(\frac{\psi_N^{n+1,k} - \psi_{N-1}^{n+1,k}}{\Delta z_{N-}} - 1 \right) + \frac{K_{N-}^{n+1,k}}{\Delta z_{N-}} \right]. \quad \text{(A17)}$$

The numerical code is written in FORTRAN 90 and is available upon request.

Competing interests. The authors declare that they have no conflict of interest.

Acknowledgements. The authors thank the anonymous referees for constructive review comments which improved the quality of the document.

Edited by: B. Berkowitz
Reviewed by: two anonymous referees

References

Bause, M. and Knabner, P.: Computation of variably saturated subsurface flow by adaptive mixed hybrid finite element methods, Adv. Water Resour., 27, 565–581, https://doi.org/10.1016/j.advwatres.2004.03.005, 2004.

Belfort, B., Younes, A., Fahs, M., and Lehmann, F.: On equivalent hydraulic conductivity for oscillation–free solutions of Richard's equation, J. Hydrol., 505, 202–217, https://doi.org/10.1016/j.jhydrol.2013.09.047, 2013.

Bergamaschi, L. and Putti, M.: Mixed finite elements and Newton-type linearizations for the solution of Richards' equation, Int. J. Numer. Methods Eng., 45, 1025–1046, https://doi.org/10.1002/(SICI)1097-0207(19990720), 1999.

Bouchemella, S., Seridi, A., and Alimi-Ichola, I.: Numerical simulation of water flow in unsaturated soils: comparative study of different forms of Richards's equation, Eur. J. Environ. Civ. Eng. 19, 1–26, https://doi.org/10.1080/19648189.2014.926294, 2015.

Celia, M. A., Bouloutas, E. T., and Zarba, R. L.: A general mass-conservative numerical solution for the unsaturated flow equation, Water Resour. Res., 26, 1483–1496, https://doi.org/10.1029/WR026i007p01483, 1990.

Clark, M. P., Fan, Y., Lawrence, D. M., Adam, J. C., Bolster, D., Gochis, D. J., Hooper, R. P., Kumar, M., Leung, L. R., Mackay, D. S., Maxwell, R. M., Shen, C., Swenson, S. C., and Zeng, X.: Improving the representation of hydrologic processes in Earth System Models, Water Resour. Res., 51, 5929–5956, https://doi.org/10.1002/2015WR017096, 2015.

Crevoisier, D., Chanzy, A., and Voltz, M.: Evaluation of the Ross fast solution of Richards' equation in unfavourable conditions for standard finite element methods, Adv. Water Resour. 32, 936–947, https://doi.org/10.1016/j.advwatres.2009.03.008, 2009.

Diersch, H.-J. G. and Perrochet, P.: On the primary variable switching technique for simulating unsaturated–saturated flows, Adv. Water Resour., 23, 271–301, https://doi.org/10.1016/S0309-1708(98)00057-8, 1999.

El Kadi, A. I. and Ling, G.: The Courant and Peclet Number criteria for the numerical solution of the Richards Equation, Water Resour. Res., 29, 3485–3494, https://doi.org/10.1029/93WR00929, 1993.

Fahs, M., Younes, A., and Lehmann, F.: An easy and efficient combination of the Mixed Finite Element Method and the Method of Lines for the resolution of Richards' Equation, Environ. Model. Softw., 24, 1122–1126, https://doi.org/10.1016/j.envsoft.2009.02.010, 2009.

Farthing, M. W., Kees, C. E., and Miller, C. T.: Mixed finite element methods and higher order temporal approximations for variably saturated groundwater flow, Adv. Water Resour., 26, 373–394, https://doi.org/10.1016/S0309-1708(02)00187-2, 2003.

Feddes, R. A., Kabat, P., Van Bakel, P. J. T., Bronswijk, J. J. B., and Halbertsma, J.: Modelling soil water dynamics in the unsaturated zone – State of the art, J. Hydrol., 100, 69–111, https://doi.org/10.1016/0022-1694(88)90182-5, 1988.

Forsyth, P. A., Wu, Y. S., and Pruess, K.: Robust numerical methods for saturated-unsaturated flow with dry initial conditions in heterogeneous media, Adv. Water Resour., 18, 25–38, https://doi.org/10.1016/0309-1708(95)00020-J, 1995.

Gottardi, G. and Venutelli, M.: UPF: two-dimensional finite-element groundwater flow model for saturated–unsaturated soils, Comput. Geosci., 27, 179–189, https://doi.org/10.1016/S0098-3004(00)00082-0, 2001.

Hills, R. G., Porro, I., Hudson, D. B., and Wierenga, P. J.: Modeling one-dimensional infiltration into very dry soils: 1. Model development and evaluation, Water Resour. Res., 25, 1259–1269, https://doi.org/10.1029/WR025i006p01259, 1989.

Hirthe, E. M. and Graf, T.: Non-iterative adaptive time-stepping scheme with temporal truncation error control for simulating variable-density flow, Adv. Water Resour., 49, 46–55, https://doi.org/10.1016/j.advwatres.2012.07.021, 2012.

Kavetski, D. and Binning, P.: Adaptive backward Euler time stepping with truncation error control for numerical modelling of unsaturated fluid flow, Int. J. Numer. Methods Eng., 53, 1301–1322, https://doi.org/10.1002/nme.329, 2002a.

Kavetski, D. and Binning, P.: Noniterative time stepping schemes with adaptive truncation error control for the solution of Richards equation, Water Resour. Res., 38, 29:1–29:19, https://doi.org/10.1029/2001WR000720, 2002b.

Kavetski, D. and Binning, P.: Truncation error and stability analysis of iterative and non-iterative Thomas–Gladwell methods for first-order non-linear differential equations, Int. J. Numer. Methods Eng., 60, 2031–2043, https://doi.org/10.1002/nme.1035, 2004.

Kavetski, D., Binning, P., and Sloan, S. W.: Adaptive time stepping and error control in a mass conservative numerical solution of the mixed form of Richards equation, Adv. Water Resour., 24, 595–605, https://doi.org/10.1016/S0309-1708(00)00076-2, 2001.

Kirkland, M. R., Hills, R. G., and Wierenga, P. J.: Algorithms for solving Richards' equation for variably saturated soils, Water Resour. Res., 28, 2049–2058, 1992.

Lehmann, F. and Ackerer, P.: Comparison of Iterative Methods for Improved Solutions of the Fluid Flow Equation in Partially Saturated Porous Media, Transp. Porous Media, 31, 275–292, https://doi.org/10.1023/A:1006555107450, 1998.

Matthews, C. J., Braddock, R. D., and Sander, G. C.: Modeling flow through a one-dimensional multi-layered soil profile using the Method of Lines, Environ. Model. Assess., 9, 103–113, 2004.

Miller, C. T., Williams, G. A., Kelley, C. T., and Tocci, M. D.: Robust solution of Richards' equation for nonuniform porous media, Water Resour. Res., 34, 2599–2610, https://doi.org/10.1029/98WR01673, 1998.

Miller, C. T., Abhishek, C., and Farthing, M. W.: A spatially and temporally adaptive solution of Richards' equation, Adv. Water Resour., 29, 525–545, https://doi.org/10.1016/j.advwatres.2005.06.008, 2006.

Mualem, Y.: A new model for predicting the hydraulic conductivity of unsaturated porous media, Water Resour. Res., 12, 513–522, https://doi.org/10.1029/WR012i003p00513, 1976.

Paniconi, C. and Putti, M.: A comparison of Picard and Newton iteration in the numerical solution of multidimensional variably saturated flow problems, Water Resour. Res., 30, 3357–3374, https://doi.org/10.1029/94WR02046, 1994.

Paniconi, C., Aldama, A. A., and Wood, E. F.: Numerical evaluation of iterative and noniterative methods for the solution of the nonlinear Richards equation, Water Resour. Res., 27, 1147–1163, 1991.

Rathfelder, K. and Abriola, L. M.: Mass conservative numerical solutions of the head-based Richards equation, Water Resour. Res., 30, 2579–2586, https://doi.org/10.1029/94WR01302, 1994.

Richards, L. A.: Capillary conduction of liquids through porous medium, J. Appl. Phys., 1, 318–333, https://doi.org/10.1063/1.1745010, 1931.

Romano, N., Brunone, B., and Santini, A.: Numerical analysis of one-dimensional unsaturated flow in layered soils, Adv. Water Resour., 21, 315–324, https://doi.org/10.1016/S0309-1708(96)00059-0, 1998.

Ross, P. J.: Modeling Soil Water and Solute Transport–Fast, Simplified Numerical Solutions, Agron. J., 95, 1352, https://doi.org/10.2134/agronj2003.1352, 2003.

Ross, P. J. and Bristow, K. L.: Simulating Water Movement in Layered and Gradational Soils Using the Kirchhoff Transform, Soil Sci. Soc. Am. J., 54, 1519, https://doi.org/10.2136/sssaj1990.03615995005400060002x, 1990.

Thomas, R. M. and Gladwell, I.: Variable-order variable-step algorithms for second-order systems, Part 1: The methods, Int. J. Numer. Methods Eng., 26, 39–53, 1988.

Tocci, M. D., Kelley, C. T., and Miller, C. T.: Accurate and economical solution of the pressure-head form of Richards' equation by the method of lines, Adv. Water Resour., 20, 1–14, https://doi.org/10.1016/S0309-1708(96)00008-5, 1997.

van Dam, J. C. and Feddes, R. A.: Numerical simulation of infiltration, evaporation and shallow groundwater levels with the Richards equation, J. Hydrol., 233, 72–85, https://doi.org/10.1016/S0022-1694(00)00227-4, 2000.

van Genuchten, M. T.: A Closed-form Equation for Predicting the Hydraulic Conductivity of Unsaturated Soils1, Soil Sci. Soc. Am. J., 44, 892, https://doi.org/10.2136/sssaj1980.03615995004400050002x, 1980.

Varado, N., Braud, I., and Ross, P. J.: Development and assessment of an efficient vadose zone module solving the 1D Richards' equation and including root extraction by plants, J. Hydrol., 323, 258–275, https://doi.org/10.1016/j.jhydrol.2005.09.015, 2006a.

Varado, N., Braud, I., Ross, P. J., and Haverkamp, R.: Assessment of an efficient numerical solution of the 1D Richards' equation on bare soil, J. Hydrol., 323, 244–257, https://doi.org/10.1016/j.jhydrol.2005.07.052, 2006b.

Vergnes, J. P., Decharme, B., Alkama, R., Martin, E., Habets, F., Douville, H.: A simple groundwater scheme for hydrological and climate applications: Description and offline evaluation over France, J. Hydrometeorol., 13, 1149–1171, 2012.

Williams, G. A., Miller, C. T., and Kelley, C. T.: Transformation approaches for simulating flow in variably saturated porous media, Water Resour. Res., 36, 923–934, https://doi.org/10.1029/1999WR900349, 2000.

Zha, Y., Shi, L., Ye, M., and Yang, J.: A generalized Ross method for two- and three-dimensional variably saturated flow, Adv. Water Resour., 54, 67–77, https://doi.org/10.1016/j.advwatres.2013.01.002, 2013.

Zha, Y., Tso, M. C.-H., Shi, L., and Yang, J.: Comparison of Non iterative Algorithms Based on Different Forms of Richards' Equation, Environ. Model. Assess., 21, 357–370, https://doi.org/10.1007/s10666-015-9467-1, 2015.