

Anonymous Referee #1

This manuscript provides a performance assessment of different known algorithms to solve the Richard's equation. In particular, the authors investigate the performance of the Ross method versus the Newton-Raphson with different time-stepping strategies. A nice set of guidelines are provided in the end. The article is well written and provides a nice contribution in this area.

We thank the referee for his/her review whose constructive comments helped in improving the manuscript. We are of course pleased that she/he considers that the manuscript presents a nice contribution to the challenging problem of solving Richards equation.

The line numbers refer to the marked manuscript.

Based on this, I suggest the publication of the manuscript after minor revision is addressed to tackle this points:

Minor comments:

The author should clearly state the assumptions in Equation (1), rigid solid matrix (negligible $\frac{d\theta}{dt}$ changes in porosity) but also need to say that $\frac{1}{\rho} \nabla \rho \approx 0$. In this context, it is worth mention that the specific storage coefficient used in Equation (1) is not exactly the same as the specific storage coefficient of the flow equation. The specific storage coefficient is the sum of compressibility of water and soil. In equation (1) the changes in porosity are neglected and therefore "so" is not exactly the specific storage. Only the part corresponding to the compressibility of water.

The text has been changed (L32-33).

Line 36: actually there are three standard forms of the equation: pressure, saturation and mixed

We do not understand this comment. In the initial manuscript, we wrote L36 'Equation (1) is also called the mixed form of RE. Two alternative formulations exist for RE' and showed the three forms of the RE (eq. 1, 2 and 3). We made some modification of the text (L39-40).

Equation (13) may be is worth to explain how to calculate fluxes q or simply refer to the appendix here for an example.

It is explained in the appendix, eq A32. We refer to it.

Equation (15), maybe is worth explaining index k

It was explained two lines later (L147 in the initial manuscript).

It is not clear whether the method suggested by Ross (2003) is mathematically equivalent to Newton-Raphson or simply performs the same way in this example. In case it is mathematically equivalent, a more detail derivation is required. In case it performs equally in

this case the manuscript should clearly state this fact. Could it be that in 2D and 3D the performance of these two algorithms are different?

The method suggested by Ross is mathematically equivalent to Newton Raphson as we explain in equation (8)-(13). There was a typo in equation (13) which may lead to misunderstanding. We corrected equation (13).

Anonymous Referee #2

Received and published: 14 March 2017

The paper compares the efficiency of a combination of two linearisation schemes for the solution of the non-linear Richards' equation with different time adaptation criteria. The first scheme is a method presented by Ross in 2003, a kind of semi-implicit scheme, calculating the non-linearities with the solution of the last time step. The second scheme is using Newton iterations. The authors first show, that if applied to the water content-formulation of Richards' equation, the method of Ross is equivalent to the first iteration of a Newton iteration. As the water-content form is only applicable to strictly unsaturated conditions, they use discretisations of the mixed form for the rest of the paper. In the Ross-type scheme, called time-adaptive (though both schemes use adaptive time stepping), the authors apply only the first-iteration of a Newton-scheme, calculating the coefficients again with the old solution, and shorten the time step until convergence. In the Newton-iteration scheme they calculate the coefficients with the last iterate until convergence. Thus in the Ross-type scheme the assembly of the linear-equation system to be solved is faster for the second or later iterates. For the adaption of the time step the authors either use an heuristic approach based on the number of Newton iterations (only for the Newton-based scheme), an approach based on an estimation of the truncation error, or a limit on the maximal allowed change of saturation. The different combinations of time-step control and linearisation approach are applied to three different test cases from the literature. The computational costs, measured in a normalized number of solves, are plotted against precision, measured as the deviation of the results from a reference solution calculated with a very fine time step and a given grid size. The authors conclude that there was no real advantage of the Ross-type scheme.

In the following, the line numbers refer to the revised marked manuscript, except when the initial manuscript is mentioned.

General comments:

The authors address a question, which has been intensively discussed in the last decades. Numerous papers on the best linearisation schemes and time-step adaptation procedures can be found easily in the literature, partially co-authored by one of the authors of this paper, many of them also cited in the paper. Thus the main question is, if the analysis of a very special scheme is a meaningful contribution to the literature and suited for publication in HESS. As there remain a lot of questions to be addressed (see specific comments below), the paper could be accepted only after major revisions. However, I am not convinced that the contributions made by the paper will be significant even after revision.

We fully agree that this question has been intensively discussed. However, we believe that the existing algorithms are still not efficient enough, especially for the recent developments of large scale models used to simulate climate change or to compute global water balances for example. It is more and more recognized that water flow in the unsaturated zone has to be modelled using mechanistic models to improve the reliability of large scale models. However, the difficulty in solving Richards equation in an efficient way (i.e. avoiding time steps in the order of minutes for simulations over several years) hampers its use in large

scale models. Therefore, we believe that there is a real need of efficient algorithms for solving Richards equation.

We added a few sentences on this motivation in the introduction (L85-L92).

Did we find THE algorithm? No, unfortunately ...

Is our contribution significant? We show for the first time that (i) the 'new' Ross method is a Newton-Raphson method, (ii) the algorithm which performs the best for each test case does not exist amongst the most existing popular algorithms and (iii) two stopping criteria have to be used instead of only one as it is implemented in many codes.

Specific comments:

equation (1) + (2): As a rigid solid matrix is assumed, s_0 could only describe the compressibility of the fluid. As water is nearly incompressible at the pressures occurring in variably saturated soils, the compressibility term is unnecessary and should be dropped.

We agree with the physical meaning of s_0 . However, from a numerical point of view, this term is very useful for solving saturated/unsaturated problems in transient. Changed in L32-L33.

line 177: "The time-adaptive algorithm consists of keeping the pressure head constant and changing the time step length." Actually, this formulation is misleading. For each tested time step a new solution for the pressure heads is calculated. Thus they are not kept constant. However, the non-linear parameters are always calculated with the solution from the old time step, corresponding to a semi-implicit scheme. Even the matrix has to be reassembled for each tested time step. Thus only the evaluation of the non-linear functions is avoided. An alternative would be the use of an interpolation table for the hydraulic functions to reduce the computational costs and still keep the accuracy high. The misleading formulation is also used in line 6 of the abstract.

We fully agree. The text has been changed (L200-L203).

line 210: I do not understand this formulation $\max_i = \left(|\Delta S_{\max}| \right)$ is the maximum of the actual change, how can it exceed itself? Do you mean exceeds $(1+\lambda)\Delta S_{\max}$?

The equation did not appear properly in the manuscript provided by HESS, sorry.

Δt^{n+1} is an estimate of the next time step. After computation over Δt^{n+1} , the saturation change can exceed the user provided S_{\max} . The text has been changed (L234-L236).

line 219-226: Is this important here? If necessary at all, please move it to the introduction

We think we have to refer to this kind of approach and we moved this part in the

introduction (L68-L77).

line 243: replace "superior to" by "larger than"

Changed L269.

line 253: "Implicit standard finite volumes" is not really a precise description. I guess you mean a cell-centred finite volume scheme for the spatial discretisation with an implicit Euler-scheme for the temporal discretisation. Actually, already in chapter 3, equation (15) the discretisation is given. Shouldn't you just refer to that section?

You are right concerning the method we used and we provide more detailed information L279-L281. However, equation (15) is more general. It also holds for other spatial discretizations like finite elements.

line 261: "the error based on the maximum change of the state variables between two iterations" would be $\max_i = \left| \psi_i^{n+1,k+1} - \psi_i^{n+1,k} \right|$. If your formula is correct you are looking at "the error base on the maximal change of the state variables in the last iteration". This actually is a very bad convergence condition as it cannot distinguish between "already converged" and "no convergence at all". However, it is also completely unclear to me, why the time truncation error should be a sensible stopping condition. A reasonable stopping condition is based on the reduction of the non-linear residual compared to the initial non-linear residual. This would really be related to a reduction of the error in the solution of the non-linear equation.

Concerning the first criterion, it is not in the last iteration (see iteration numbered by k). It is during the iterative process. If this criterion is met, the process is stopped and the computation of the next time step is performed. This is a very popular stopping method, not only for unsaturated flow but also for density driven flow for example.

Time stopping criteria have been applied by others (see references in the manuscript).

Residuals are also used as stopping criterion but it performs like the criterion based on the maximum change of the state variable. Both criteria are linearly linked (Ackerer et al. 1999 Modeling Variable Density Flow and Solute Transport in Porous Medium: 1. Numerical Model and Verification. Transport in Porous Media 35: 345–373).

We add the following in the revised manuscript (L299-L302):

“We also analyzed convergence based on the non-linear residual. It was found 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 non-linear residuals are not reported.”

equation (27) and (28): is it really necessary to write out this equations? Is it not enough to state that relative and absolute error bounds are given?

The tolerance values can be different. We provide more details (L291-L296).

line 269-271: Actually, not all possible combinations have been performed. You could also have tested using only the truncation error (if this makes sense).

We make all possible combinations but we reported only the most significant results in the paper as stated in L270-L271 in the initial manuscript.

line 293-294: as the spatial discretisation error (though not explicitly mentioned) is addressed here: How did you check, that the grid really was fine enough? As you try to get very accurate solutions in time (down to an error of 10^{-5}), did you really make sure, that the grid is fine enough to produce changes significantly lower than 10^{-5} if further refined?

We did it in the traditional way, by successive grid refinements. See below, discussion on figures 4,6,8... We added some information L324-L326.

line 301-303: As you are using a mixed scheme: why did you not just calculate ΔS_{\max} from the saturations? I am also a bit confused about notation. In equation (20) S_{\max} was a "user-defined maximum saturation change", now it is something calculated from the solution...

We want to compare methods that use different criteria to stop the iterative procedure. Equation (29) gives the relationship between both user's defined criteria.

We changed the text L334-L336.

line 306-315: If the mixed form of Richards' equation is used, with a (locally mass-conservative) finite volume discretisation and the linear equations are solved sufficiently accurate, why should there be mass balance at all? It is obvious from the beginning that this could only hint to errors in your code. Thus the statement in line 314-315 is trivial.

We agree. We mentioned that the mass balance errors were negligible and our comparisons are not based on this error. We just check this error because it is commonly used to compare numerical schemes (even the codes which solve mixed form of RE).

line 328-330: I do not understand, why the computational costs of the time-adaptive algorithm are calculated by $(N_{sol}+N_{param})/2$. For each iteration step in the iterative scheme you have to calculate the nonlinear parameters and their derivatives, assemble a matrix and solve a linear equation system. For each iteration step in the time-adaptive scheme you have to assemble a matrix and solve a linear equation system, while you have to calculate the nonlinear parameters and their derivatives only once for each time step. So the cost reduction depends on the number of iterations necessary (if it is always one iteration, there is no cost reduction at all) and on the relative computational cost of nonlinear parameter evaluation

compared to assembly and solution of the linear equation system. Why should this result just in this simple formula?

We assume that the computational costs are depending on the time required to compute the non-linear parameters and the time required to solve the system of equations i.e. N_{param} , the number of calls of the subroutine which computes the parameter values and N_{sol} the number of calls to solve the equations. For the standard approach, N_{param} is equal to N_{sol} . So we have computational costs that are equal to $2N_{\text{sol}}$ for the standard approach and $(N_{\text{sol}}+N_{\text{param}})$ for the time adaptive scheme. This is why we used N_{sol} for the standard approach and $(N_{\text{sol}}+N_{\text{param}})/2$ for the time adaptive scheme.

We added a sentence in the text (L362).

figure 4, 6 and 8: for the two saturation-based schemes which allow the highest precision in all three scenarios, there is often a reduced increase of precision with costs at high precision. This could be a hint that the spatial resolution was not high enough and that in this cases the spatial discretisation error became relevant. I would thus not agree with the conclusions in line 368-371.

We agree on the reduced increase of precision and we do not have any clear explanation. The grid size is the same for all schemes, so we assume that the error due to the spatial discretization is the same for all schemes.

We redo the computation of TC1 with a spatial discretization two times finer and relative tolerances of 10^{-4} and 10^{-5} . The differences could not be seen on the profile (see figure below obtained for a tolerance of 10^{-5}). Therefore, we disagree with the reviewer's statement; the spatial discretization is high enough.

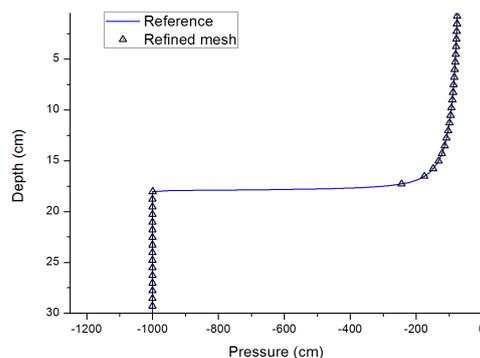


figure 4, 6 and 8: there is something strange with all the figures. While in the tables there are only values for four precisions given, there are always six points in the figures for the truncation based algorithms but only four points for all other algorithms. This does not make sense.

Tables with too many numbers are boring. We provide more point in figures to better underline the trends which are difficult to see in tables. It is explained L332-L333 in the initial manuscript.

line 342-348: As both stopping criteria for the non-linear iterations are not very adequate and a condition based on the reduction of non-linear defect should be used, I will not comment on the comparison of this non-adequate criteria.

The criteria are quite popular and, for our examples, more restrictive than residual based stopping criteria. We add some comments L299-L302.

line 372-375: I do not agree with the last statement. As the saturation based time stepping TA_S already produced the same precision when a precision of 10^{-4} was demanded, it also had a comparable efficiency with the truncation error based algorithm for this case. The only problem was, that the error was not reduced with the higher precision, probably linked to a not fine enough spatial grid. A not mentioned point is, that for the saturation based time step control, there was a linear decrease of the error with the specified precision, whereas this was more erratic for the truncation based time step control.

As already mentioned, it is not a problem of spatial discretization.

We mentioned the linear decrease L409-L412.

line 398-401: I do not understand this statement. After all, the algorithm did compute a solution, so why was the time step too long for reaching convergence? And if it did not reach convergence, how could it calculate the next time steps?

We revised figure 7 and changed the text accordingly. There was some mixed up use of the data files. We checked all computations and dataset. Thanks for pointing out this error. We changed the comments (L432-442).

line 405-407: Actually, the first two scenarios also had a step change of boundary conditions at the beginning and thus a "non-monotonic" change of boundary conditions. Thus this is not really completely different

Table 2 clearly indicates that the boundary conditions are not changing in time for the first two scenarios.

line 410: "to avoid a too rough discretisation of the upper boundary conditions": did you make sure that the times at which the boundary condition changed were reached exactly? If you did not do this, you get unnecessarily wrong solutions. This is not a question of the time stepping strategy, but of common sense and not difficult to implement. As I do not

know if this was done, I am not going to discuss the further results of test case 3.

The upper boundary fluxes change with a time step of 1 day (see Fig. 9). To well describe this time varying boundary conditions, we fixed the maximum time step length to 0.2 day.

For a given time, the boundary conditions are linearly interpolated. We explained this issue more properly L451-L455.

line 447-449: this also means that most of the algorithms are not really suitable for error control. The relation between specified precision and obtained error is not linear for most of the algorithms.

We agree.

line 450-452: This is a trivial remark as a locally mass conservative discretisation scheme is used. It would be different for e.g. standard finite-elements as used in Hydrus.

It is trivial for the experts and it would not be different for finite elements schemes which also preserve mass if the mass balance is computed consistently with the method i.e. on the dual mesh.

line 453-456: What should really be implemented is a convergence condition based on a reduction of the non-linear residual.

As stated previously, the stopping criteria we used are more restrictive.

line 457-460: This should be formulated much clearer: The time-adaptive algorithm with the truncation based time-stepping condition did fail to produce accurate results for almost all test cases and converged to the wrong result in the first test cases. Thus it is useless. I would not expect that this will change for 2D or 3D problems. With the saturation-based time-stepping, the time-adaptive algorithm was overall comparable to the standard iterative approach. However, it always was rather costly at high precision, where the time steps are small and thus the number of iterations per time step was also small. Thus the advantage of not calculating the non-linear parameters did not pay off. This also should be similar for 2D and 3D calculations.

Time adaptive performed quite well for the test TC2 (see fig. 6). The difference between our 1D and 2-3D calculations is of course the number of elements and therefore the number of time the parameters have to be computed. Therefore, for a given accuracy, the time adaptive algorithm might be more efficient. We re-wrote the conclusion (L494-L525).

line 462-468: I still do not get, why the time truncation error should be a relevant

stopping condition for the non-linear iterations within one time step. Obviously the maximal change of the potential alone is not a reasonable condition, as it is linked to the fluxes and saturation changes via highly non-linear functions.

Our study shows that the time truncation error is a relevant criterion. The three test cases that we have done show the relevance of the algorithm that uses the time truncation error.

We re-wrote the conclusion (L494-L525).

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

Hassane^{1,2} ~~Mamadou~~ Maina F.-Z., Ackerer^{1,*} P.

¹Laboratoire d'Hydrologie et de Géochimie de Strasbourg,
Univ. Strasbourg/EOST - CNRS
1 rue Blessig, 67084 Strasbourg,
France

²CEA-Laboratoire de Modélisation des Transferts dans l'Environnement,
Bât. 225, F-13108 Saint Paul lez Durance cedex,
France

*Corresponding author: ackerer@unistra.fr

ABSTRACT

1 The solution of the mathematical model for flow in variably saturated porous media described
2 by Richards equation (RE) is subject to heavy numerical difficulties due to its highly non-
3 linear properties and remains very challenging. Two different algorithms are used in this work
4 to solve the mixed-form of RE: the traditional iterative algorithm and a time-adaptive
5 algorithm consisting of changing the time step magnitude within the iteration procedure while
6 ~~the non-linear parameters are computed with the state variable at the previous time, the state~~
7 ~~variable is kept constant.~~ The Ross method is an example of this type of scheme, and we show
8 that it is equivalent to the Newton-Raphson method with a time-adaptive algorithm.
9 Both algorithms are coupled to different time stepping strategies: the standard heuristic
10 approach based on the number of iterations and two strategies based on the time truncation
11 error or on the change of water saturation. Three different test cases are used to evaluate the
12 efficiency of these algorithms.

13 The numerical results highlight the necessity of implementing ~~two types of errors: the~~
14 ~~iterative convergence error (maximum difference of the state variable between two iterations)~~
15 ~~and~~ an estimate of the time truncation errors. ~~The algorithms using these two types of errors~~
16 ~~together were found to be the most efficient when highly accurate results are required.~~

17

18 **Key words:** Unsaturated flow, Newton-Raphson, Time stepping

19

20 **1. Introduction**

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

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

31
$$\begin{cases} \frac{\partial \theta}{\partial t} + S_w s_0 \frac{\partial \psi}{\partial t} + \nabla \cdot \mathbf{q} = f \\ \mathbf{q} = -k_r(\psi) \mathbf{K} [\nabla \psi + \nabla z] \end{cases} \quad (1)$$

Code de champ modifié

32 where θ is the volumetric water content [L^3/L^3], S_w is the water saturation [-], s_0 ~~is the specific~~
33 ~~storage coefficient accounts for fluid compressibility~~ [L^{-1}], ψ is the pressure head [L], \mathbf{q} is the
34 water flux based on the extended Darcy's law [L/T], t is the time [T], z is the vertical
35 coordinate (positive upward) [L], f is the sink/source term [T^{-1}], \mathbf{K} is the saturated hydraulic
36 conductivity tensor [L/T] and $k_r(\psi)$ is the relative hydraulic conductivity [-]. The model
37 includes initial and boundary conditions of the Dirichlet (prescribed pressure head) or
38 Neumann (prescribed flux) type.

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

41 The pressure form is defined by:

42

$$\begin{cases} [C(\psi) + S_w S_0] \frac{\partial \psi}{\partial t} + \nabla \cdot \mathbf{q} = f \\ \mathbf{q} = -k_r(\psi) \mathbf{K} [\nabla \psi + \nabla z] \end{cases} \quad (2)$$

Code de champ modifié

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

45

$$\begin{cases} \frac{\partial \theta}{\partial t} + \nabla \cdot \mathbf{q} = f \\ \mathbf{q} = -(\mathbf{D}(\theta) \nabla \theta + k_r(\theta) \mathbf{K} \nabla z) \end{cases} \quad (3)$$

Code de champ modifié

46 where $\mathbf{D}(\theta) = k_r(\theta) \mathbf{K} \frac{d\psi}{d\theta}$ is the pore water diffusivity [L^2/T].

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

49

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

Code de champ modifié

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

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

54

$$k_r(S_w) = \begin{cases} S_w^{1/2} [1 - (1 - S_w^{1/m})^m]^2 & \psi < 0 \\ 1.0 & \psi \geq 0 \end{cases} \quad (5)$$

Code de champ modifié

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

56 Due to the strong heterogeneities of the unsaturated zone and nonlinearities in the constitutive
57 relations (Eq. (4)(4) and (5)(5)), analytical solution of RE does not exist except in special

Mis en forme : Anglais (États Unis)

Mis en forme : Anglais (États Unis)

58 cases (Celia et al., 1990; van Dam and Feddes, 2000). Therefore, numerical methods such as
59 finite difference (Feddes et al., 1988; Romano et al., 1998; van Dam and Feddes, 2000), finite
60 element (Gottardi and Venutelli, 2001), and mixed finite element (Bause and Knabner, 2004;
61 Bergamaschi and Putti, 1999; Fahs et al., 2009; Farthing et al., 2003) are used to solve RE.

62 Iterative methods based on the Picard (fixed point) or Newton-Raphson approach (Lehmann
63 and Ackerer, 1998; Paniconi and Putti, 1994) are the most popular techniques for solving this
64 highly nonlinear equation. Alternative iterative methods are based on transform formulations
65 (Crevoisier et al., 2009; Ross and Bristow, 1990; Williams et al., 2000; Zha et al., 2013) or
66 the method of lines (Fahs et al., 2009; Matthews et al., 2004; Miller et al., 1998; Tocci et al.,
67 1997).

68 Adaptive time stepping strategies based on time truncation error control were found to be
69 superior to others approaches (Hirthe and Graf, 2012; Kavetski et al., 2001; Tocci et al.,
70 1997). The Method of Lines using the DASPCK integrator was applied to the Richards'
71 equation by Matthews et al. (2004), Miller et al. (1998), Tocci et al. (1997) among others. The
72 Method of Lines consists of discretization of the spatial part of the PDE only, leading to a
73 system of ordinary differential equations. It has been found to be significantly more efficient
74 than other temporal discretizations (Miller et al., 2006). However, Kavetski and Binning
75 (2002b) reported difficulties in obtaining convergence for the DASPCK solver associated with
76 an arithmetic mean of inter-block conductivities for the most difficult problem addressed by
77 Miller et al. (1998).

78 Additionally, very few non-iterative schemes have been developed (Kavetski and Binning,
79 2004, 2002a; Paniconi et al., 1991).

80

81 Despite the many existing numerical methods, solution of the RE is still a challenging
82 research topic with many remaining questions about reduction of the computational time,
83 treatment of nonlinearities, and improvement of the accuracy of these methods for difficult
84 problems such as infiltration in very dry soils (Diersch and Perrochet, 1999; Forsyth et al.,
85 1995; R. G. Hills, 1989). The need of efficient algorithms for solving this equation has
86 increased during the last decades because it has been recognized that explicit modelling of
87 flow in the unsaturated zone has to be implemented in Land Surface Models (Vergnes et al.,
88 2012). In their recent review of land surface models, Clarke et al. (2015) push for a
89 mechanistic modelling of the flow in soils. They consider that the implementation of the
90 mixed form of the Richards equation is an improvement of the modeling of soil moisture
91 variations. They also underline the need of efficient algorithms for solving the RE to allow the
92 implementation of stochastic approaches and/or automatic parameter estimations.

93 In this study, we analyzed the performance of different algorithms based on the Newton-
94 Raphson method since the classical Picard scheme has been found less efficient (Lehmann
95 and Ackerer, 1998). Applied to the soil moisture form of the RE equation, we demonstrate
96 that the recently developed Ross method (Ross, 2003; Crevoisier et al., 2009; Zha et al., 2013)
97 is equivalent to Newton-Raphson method (section 2). A detailed presentation of the Newton-
98 Raphson method applied to the mixed form or RE is given in section 3. The standard Newton-
99 Raphson algorithm is based on the computation of the corresponding matrices in an iterative
100 way by updating the parameters until convergence. An alternative algorithm has been
101 suggested more recently where the parameters are kept unchanged within one time step and
102 the time step is adapted to reach convergence. This algorithm has been applied to the
103 pressure-based form of RE by Kavetski and Binning (2002a) and to the soil moisture form by
104 Crevoisier et al. (2009), Ross (2003), Zha et al. (2013). Although this algorithm is called “non
105 iterative” because the parameters are not updated during the calculation, iterations may be

106 necessary to adapt the magnitude of the time step. Therefore, in the following, we will refer to
 107 the usual algorithm as “iterative” and to the alternative algorithm as “time-adaptive”. To our
 108 knowledge, this alternative algorithm has never been applied to the mixed form of RE.
 109 Section 4 is dedicated to both algorithms and to the time stepping strategy used for solving
 110 RE. Finally, in section 5, the numerical accuracy and robustness of the algorithms applied to
 111 the mixed-form of RE are evaluated using three different test cases.

112

113 2. The Ross method and the Newton-Raphson method

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

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

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

$$124 \quad \frac{\Delta z}{\Delta t} (\theta_i^{n+1} - \theta_i^n) = \frac{\Delta z}{\Delta t} (\theta_{s,i} - \theta_{r,i}) (S_i^{n+1} - S_i^n) = q_-^\sigma - q_+^\sigma \quad (6)$$

125 with:

$$126 \quad \begin{cases} q_+^\sigma = q_+^n + \sigma \left[\left(\frac{\partial q_i^n}{\partial S_i^n} \right) (S_i^{n+1} - S_i^n) + \left(\frac{\partial q_i^n}{\partial S_{i+1}^n} \right) (S_{i+1}^{n+1} - S_{i+1}^n) \right] \\ q_-^\sigma = q_-^n + \sigma \left[\left(\frac{\partial q_i^n}{\partial S_i^n} \right) (S_i^{n+1} - S_i^n) + \left(\frac{\partial q_i^n}{\partial S_{i-1}^n} \right) (S_{i-1}^{n+1} - S_{i-1}^n) \right] \end{cases} \quad (7)$$

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

Code de champ modifié

Code de champ modifié

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

$$132 \quad -\left(\frac{\partial q_-^n}{\partial S_{i-1}^n}\right)(S_{i-1}^{n+1} - S_{i-1}^n) + \left[\frac{\Delta z}{\sigma \Delta t}(\theta_{s,i} - \theta_{r,i}) - \left(\left(\frac{\partial q_-^n}{\partial S_i^n}\right) - \left(\frac{\partial q_+^n}{\partial S_i^n}\right)\right)\right](S_i^{n+1} - S_i^n) \\ 133 \quad + \left(\frac{\partial q_+^n}{\partial S_{i+1}^n}\right)(S_{i+1}^{n+1} - S_{i+1}^n) = q_-^n - q_+^n \quad (8)$$

Code de champ modifié

134 The Newton-Raphson method was initially developed as a root-finding algorithm of an
135 arbitrary equation that has been generalized for solving a system of non-linear equations.
136 Applied to the soil moisture form of the RE and using an implicit scheme, the NR consists in
137 defining a residual based on the mass balance equation (Eq. (6)) at iteration k for time step
138 $n+1$ and for cell i written as:

$$139 \quad R_i^{n+1,k} = \frac{\Delta z}{\Delta t}(\theta_{s,i} - \theta_{r,i})(S_i^{n+1,k} - S_i^n) + q_+^{n+1,k} - q_-^{n+1,k} \quad (9)$$

Code de champ modifié

140 where $R_i^{n+1,k}$ is called the residual.

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

$$143 \quad \frac{R_i^{n+1,k}}{\partial S_{i-1}^{n+1,k}}(S_{i-1}^{n+1,k+1} - S_{i-1}^{n+1,k}) + R_i^{n+1,k} = 0 \quad (10)$$

Code de champ modifié

144 The derivatives of this residual are:

$$145 \quad \begin{cases} \frac{\partial R_i^{n+1,k}}{\partial S_{i-1}^{n+1,k}} = -\frac{\partial q_-^{n+1,k}}{\partial S_{i-1}^{n+1,k}} \\ \frac{\partial R_i^{n+1,k}}{\partial S_i^{n+1,k}} = \frac{\Delta z}{\Delta t}(\theta_{s,i} - \theta_{r,i}) + \frac{\partial q_+^{n+1,k}}{\partial S_i^{n+1,k}} - \frac{\partial q_-^{n+1,k}}{\partial S_i^{n+1,k}} \\ \frac{\partial R_i^{n+1,k}}{\partial S_{i+1}^{n+1,k}} = \frac{\partial q_+^{n+1,k}}{\partial S_{i+1}^{n+1,k}} \end{cases} \quad (11)$$

Code de champ modifié

146
147 which leads to the following set of linear equations:
148

149

$$-\frac{\partial q_-^{n+1,k}}{\partial S_{i-1}^{n+1,k}}(S_{i-1}^{n+1,k+1} - S_{i-1}^{n+1,k}) + \left[\frac{\Delta z}{\Delta t}(\theta_{s,i} - \theta_{r,i}) + \frac{\partial q_+^{n+1,k}}{\partial S_i^{n+1,k}} - \frac{\partial q_-^{n+1,k}}{\partial S_i^{n+1,k}} \right] (S_i^{n+1,k+1} - S_i^{n+1,k})$$

150

$$+ \frac{\partial q_+^{n+1,k}}{\partial S_{i+1}^{n+1,k}}(S_{i+1}^{n+1,k+1} - S_{i+1}^{n+1,k}) = \frac{\Delta z}{\Delta t}(\theta_{s,i} - \theta_{r,i})(S_i^{n+1,k} - S_i^n) + q_+^{n+1,k} - q_-^{n+1,k}$$

(12)

Code de champ modifié

151 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 :

152

$$-\frac{\partial q_-^n}{\partial S_{i-1}^n}(S_{i-1}^{n+1} - S_{i-1}^n) + \left[\frac{\Delta z}{\Delta t}(\theta_{s,i} - \theta_{r,i}) + \frac{\partial q_+^n}{\partial S_i^n} - \frac{\partial q_-^n}{\partial S_i^n} \right] (S_i^{n+1} - S_i^n)$$

153

$$+ \frac{\partial q_+^n}{\partial S_{i+1}^n}(S_{i+1}^{n+1} - S_{i+1}^n) = q_+^n - q_-^n$$

(13)

Code de champ modifié

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

159 3. Newton Raphson method for the mixed form Richards' equation

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

164 The mixed form of the Richards equation given by equation (1) is rewritten as:

165

$$\frac{\partial \theta}{\partial t} + S_w S_0 \frac{\partial \psi}{\partial t} = \nabla \cdot k_r(\psi) \mathbf{K} [\nabla \psi + \nabla z] + f$$

(14)

Code de champ modifié

166 and is discretized by:

167

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

(15)

Code de champ modifié

168 where \mathbf{A} is the discretized form of the divergence term, \mathbf{B} and \mathbf{E} are the discretized forms of
169 the storage terms and \mathbf{F} is the discretized form of the sink/source term and the boundary

170 conditions, n is the time step and k the iteration counter. Δt^{n+1} is the time step magnitude
 171 defined by $\Delta t^{n+1} = t^{n+1} - t^n$. Matrices \mathbf{A} , \mathbf{B} , \mathbf{E} and vector \mathbf{F} depend on the numerical scheme
 172 used for the spatial discretization. The implicit scheme is applied for the spatial discretization.
 173 For the Newton-Raphson method, the residual is defined now by:

$$174 \quad \mathbf{R}(\psi^{n+1,k}) = \mathbf{A}^{n+1,k} \psi^{n+1,k} + \mathbf{B}^{n+1,k} \frac{\psi^{n+1,k} - \psi^n}{\Delta t^{n+1}} + \mathbf{E} \frac{\theta^{n+1,k} - \theta^n}{\Delta t^{n+1}} - \mathbf{F}^{n+1,k} \quad (16)$$

175 and its derivatives are:

$$176 \quad \mathbf{R}'(\psi^{n+1,k}) = \mathbf{A}^{n+1,k} + \frac{\partial \mathbf{A}^{n+1,k}}{\partial \psi^{n+1,k}} \psi^{n+1,k} + \frac{\mathbf{B}^{n+1,k}}{\Delta t^{n+1}} + \frac{\partial \mathbf{B}^{n+1,k}}{\partial \psi^{n+1,k}} \frac{\psi^{n+1,k} - \psi^n}{\Delta t^{n+1}} \\ + \frac{\mathbf{E}}{\Delta t^{n+1}} \frac{\partial \theta^{n+1,k}}{\partial \psi^{n+1,k}} - \frac{\partial \mathbf{F}^{n+1,k}}{\partial \psi^{n+1,k}} \quad (17)$$

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

$$178 \quad \mathbf{R}'(\psi^{n+1,k}) \Delta \psi^{n+1,k+1} = -\mathbf{R}(\psi^{n+1,k}) \quad (18)$$

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

180
 181 The NR formulation is also used for the non-iterative scheme by applying only one NR step
 182 per time step, with $\psi^{n+1} = \psi^{n+1,1}$ where $\psi^{n+1,0} = \psi^n$ (Paniconi et al., 1991; Zha et al., 2015).

183

184 4. Algorithms and time stepping strategy

185 The usual algorithm used to solve RE consists in defining a time step that remains constant
 186 and to iteratively compute the parameters and variables in the following way:

187 **For a given time step n**

- 188 - Define the time step length Δt^{n+1} depending on the time stepping strategy.
- 189 - Initialization of the iterative process by setting $\psi^{n+1,1} = \psi^n$.

190 *do $k=1$, maxit*

191 1. Computation of the variable $\theta^{n+1,k}$, the parameter $\mathbf{K}^{n+1,k}$ and their derivatives

192
$$\frac{d\theta^{n+1,k}}{d\psi^{n+1,k}}, \frac{\partial \mathbf{K}^{n+1,k}}{\partial \psi^{n+1,k}} \text{ using } \psi^{n+1,k}.$$

193 2. Computation of the system matrix \mathbf{R}' and the residual \mathbf{R} .

194 3. Computation of the system solution $\psi^{n+1,k+1}$.

195 4. Check convergence. If convergence is achieved, exit.

196 *enddo*

197 ***Next time step***

198 where k is the iteration counter and *maxit* the maximum number of iterations.

199

200 The time-adaptive algorithm consists in calculating the non-linear parameters with the
201 pressure heads computed at time step n and adapting the time step length, of keeping the
202 pressure head constant and changing the time step length. The algorithm is described by the
203 following:

204

205 ***For a given time step n***

206 - Computation of the variable θ^n , the parameter \mathbf{K}^n and their derivatives

207
$$\frac{d\theta^n}{d\psi^n}, \frac{\partial \mathbf{K}^n}{\partial \psi^n} \text{ using } \psi^n.$$

208 *do k=1, maxit*

209 1. Define a time step $\Delta t^{n+1,k}$ depending on the time stepping strategy.

210 2. Computation of the system matrix \mathbf{R}' and the residual \mathbf{R} .

211 3. Computation of the system solution $\psi^{n+1,k+1}$.

212 4. Check convergence. If convergence is achieved, exit.

213 *enddo*

214 ***Next time step***

215

216 The main advantage of the alternative algorithm is its avoidance of the computation of the
217 variable θ , the parameter \mathbf{K} and their derivatives $\frac{d\theta}{d\psi}$ and $\frac{\partial \mathbf{K}}{\partial \psi}$ during the iterations. Due to

218 the highly nonlinear relations between θ , \mathbf{K} , $\frac{d\theta}{d\psi}$, $\frac{\partial \mathbf{K}}{\partial \psi}$ and the pressure, this computation
 219 may require significant CPU time.

220
 221 The most popular time step management during the simulation is that of the heuristic type
 222 (Miller et al., 2006). The time step Δt^{n+1} is computed depending on Δt^n and the number of
 223 iterations k necessary to reach convergence in the following way:

$$224 \quad \begin{cases} \text{if } k \leq m_1 & \Delta t^{n+1} = k_1 \Delta t^n & k_1 > 1.0 \\ \text{if } m_1 \leq k \leq m_2 & \Delta t^{n+1} = \Delta t^n \\ \text{if } m_2 \leq k & \Delta t^{n+1} = k_2 \Delta t^n & k_2 < 1.0 \end{cases} \quad (19)$$

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

226
 227 Other heuristic time step management procedures have been suggested by Kirkland et al.,
 228 (1992) based on the water volumes exchanged between the adjacent cells of the grid and by
 229 Ross (2003), where the time step size is controlled by the maximum allowed change in the
 230 saturation.

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

$$233 \quad \Delta t^{n+1} = \frac{\Delta S_{max}}{\max_i \left(\frac{|q_{-i}^n - q_{+i}^n|}{\Delta z_i (\theta_{s,i} - \theta_{r,i})} \right)} \quad (20)$$

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

$$238 \quad \Delta t^{n+1,k} = \frac{\Delta S_{max}}{\max_i (|\Delta S_i|)} \Delta t^{n+1,k-1} \quad (21)$$

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

242
 243 ~~Adaptive time stepping strategies based on time truncation error control were found to be~~
 244 ~~superior to others approaches (Hirthe and Graf, 2012; Kavetski et al., 2001; Tocci et al.,~~

Code de champ modifié

Code de champ modifié

Code de champ modifié

1997). The Method of Lines using the DASPK integrator was applied to the Richards' equation by Matthews et al. (2004), Miller et al. (1998), Tozzi 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).

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:

$$\Psi_{(1)}^{n+1} = \Psi^n + \Delta t^{n+1} \frac{\partial \Psi^n}{\partial t} \quad (22)$$

Code de champ modifié

The second-order approximation is:

$$\begin{aligned} \Psi_{(2)}^{n+1} &= \Psi^n + \Delta t^{n+1} \frac{\partial \Psi^n}{\partial t} + \frac{1}{2} (\Delta t^{n+1})^2 \frac{\partial^2 \Psi^n}{\partial t^2} \\ &= \Psi^n + \frac{1}{2} (\Delta t^{n+1}) \left[\frac{\partial \Psi^{n+1}}{\partial t} + \frac{\partial \Psi^n}{\partial t} \right] \end{aligned} \quad (23)$$

Code de champ modifié

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

This truncation error is given by:

265

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

Code de champ modifié

266 When the truncation error is smaller than γ , the temporal truncation error tolerance defined by
 267 the user, the size of the next time step is calculated by:

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

Code de champ modifié

269 When the truncation error is ~~superior to~~ larger than γ , the computation is repeated with a
 270 reduced time step defined as following:

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

Code de champ modifié

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

276

277 5. Evaluation of the algorithms' performance

278 We applied the NR method to the mixed form of RE using the standard iterative algorithm
 279 and the time-adaptive algorithm. A cell centered finite volume scheme for the spatial
 280 discretization with an implicit Euler-scheme for the temporal discretization has Implicit
 281 standard finite volumes have been used to solve the partial differential equation and
 282 arithmetic means are used to compute the inter-block hydraulic conductivity. The detailed
 283 discretizations of the matrix $\mathbf{R}'(\psi^{n+1,k})$ and the vector $\mathbf{R}(\psi^{n+1,k})$ (see Eq.-(18)) are given in

284 Appendix 1. The time-adaptive algorithms have been applied as described by the authors:
285 Ross (2003) for the time stepping based on the saturation changes and Kavetski et al. (2001)
286 for the time stepping based on the truncation errors.

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

291
$$\varepsilon_\psi < \tau_{\psi,a} + \tau_{\psi,r} |\psi_{imax}^{n+1,k+1}| \quad (27)$$

Code de champ modifié

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

294
$$\varepsilon_t < \tau_{t,a} + \tau_{t,r} |\psi_{imax}^{n+1,k+1}| \quad (28)$$

Code de champ modifié

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

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

303 We investigated three one-dimensional problems with various initial and boundary conditions
304 and hydraulic functions to assess the accuracy, efficiency and computational costs of the

305 different algorithms. The selected test cases represent a range of difficult infiltration problems
306 widely analyzed in the literature:

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

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

Mis en forme : Anglais (États Unis)

Mis en forme : Anglais (États Unis)

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

324 Preliminary tests were performed to define the optimal spatial discretization i.e. a finer spatial
325 discretization provided very similar results for a given convergence criterion and a given time
326 stepping strategy. Therefore, ~~W~~we can assume that the errors are only originated from the
327 time step size and the linearization.

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

329 - $k_1=0.80, k_2=1.20, m_1=5, m_2=10$, which are the usual values for the heuristic strategy
 330 defined by Eq. (19)(19);
 331 - $r_{min}=0.10, r_{max}=4.0, s=0.9, EPS=10^{-10}$, which are the standard values for the time
 332 stepping scheme based on time discretization error defined by Eq. (26)(26) (Kavetski
 333 et al., 2001);

Mis en forme : Police :Times New Roman, 12 pt

Mis en forme : Police :(Par défaut) Times New Roman, Anglais (États Unis)

334 To perform a consistent comparison of the time step strategies, the maximum allowed change
 335 in saturation (see equation (20) and (21)) has been evaluated using the maximum change in
 336 the pressure, according to the following relationship:

$$337 \quad \Delta S_{max} \approx \frac{1}{(\theta_{s,imax} - \theta_{r,imax})} \frac{d\theta}{d\psi} \Big|_{\psi_{imax}} \left(\tau_a + \tau_r \left| \psi_{imax}^{n+1,k+1} \right| \right) \quad (29)$$

Code de champ modifié

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

339

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

$$342 \quad MB(t^{n+1}) = \frac{\sum_{i=1}^M \Delta z_i (\theta_i^{n+1} - \theta_i^0)}{\sum_{k=1}^{n+1} (q_{in}^k - q_{out}^k) \Delta t^k} \quad (30)$$

Code de champ modifié

343 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
 344 initial water content, and q_{in}^k and q_{out}^k are the inflow and outflow, respectively, at the domain
 345 boundaries at time t^k . M is the number of cells/elements. The fluxes at the boundaries are
 346 defined by $q^k = \frac{1}{2}(q^k + q^{k-1})$. The mass balance errors were checked for each runs but were
 347 found to be negligible since we solved the mass-conserving RE form.

348 While it is necessary to satisfy the global mass balance for an accurate numerical scheme, a
 349 low mass balance error is not sufficient to ensure the accuracy of the solution. Therefore,
 350 solutions have also been compared with the reference solution obtained using a very fine
 351 temporal discretization and the iterative Newton-Raphson method. This comparison is based
 352 on the average relative error defined by:

$$353 \quad \varepsilon_k = \left[\frac{1}{M} \sum_i \frac{|\psi_i^{ref} - \hat{\psi}_i|^k}{|\psi_i^{ref}|^k} \right]^{1/k} \quad (31)$$

Code de champ modifié

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

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

369

370 *TC1: Infiltration in a homogenous soil with constant boundary conditions*

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

377 The results for the iterative and time-adaptive algorithms are presented in Tables 3 and 4,
378 respectively. When both convergence criteria are used (algorithms SH $_{\Delta\psi_{\Delta t}}$ and
379 SS $_{\Delta\psi_{\Delta t}}$), N_{trunc} represents the number of times where the truncation error is the most
380 restrictive condition. For the heuristic time stepping schemes, the convergence is mostly
381 linked to the truncation error (N_{trunc} is close to N_{sol}), whereas when the saturation time
382 stepping scheme is used, the most restrictive criterion is the maximum difference in the
383 pressure.

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

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

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

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

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

407 For this test case, the most efficient algorithms are the iterative algorithms using the time
408 stepping strategy based on truncation error (ST_ $\Delta\psi$) or based on the saturation changes
409 (SS_ $\Delta\psi$ _ Δt). Saturated based time stepping strategies (SS_ $\Delta\psi$ _ Δt and TA_S) shows a linear
410 decrease of L_2 with computational costs, except for the case of For very high precision ($L_2 <$
411 10^{-4}), where ST_ $\Delta\psi$ outperforms the other algorithms. No convincing explanation has been
412 found for the insignificant change in accuracy for SS_ $\Delta\psi$ _ Δt at high precision. :

413

414 *TC2: Infiltration in a homogenous soil with hydrostatic initial conditions*

415 This test case models an infiltration in a 200 cm vertical column of unconsolidated clay loam
416 with non-uniform grain size distribution and was considered by Miller et al. (1998) to be a
417 very challenging test. This problem was found to be more challenging from the numerical

418 point of view compared to TC1 due to the relative permeability function that enhances the
419 non-linear behavior of Richards' equation (Fig. 1, 2, 3). The cell size has been set to 0.125
420 cm, the initial time step to 10^{-5} s and the maximum time step magnitude to 1000 s.

421 The different norms for the iterative and the time-adaptive schemes are given in Tables 5 and
422 6.

423 Investigation of this test case leads to similar qualitative conclusions when the time stepping
424 scheme is based on the saturation differences (SS_Δψ_Δt and TA_S). The standard scheme
425 SH_Δψ fails to provide an accurate solution within a reasonable number of iterations (less
426 than 10^7).

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

430 Figure 7 shows the time step magnitudes for approximately equal L_2 -norms for the two time-
431 adaptive algorithms and for the iterative algorithm using truncation errors for time stepping
432 ($4.254 \cdot 10^{-4}$ within 3503 iterations for ST_Δψ, $4.563 \cdot 10^{-4}$ within ~~3094~~3098 iterations for
433 TA_T and $4.844 \cdot 10^{-4}$ within 113583 iterations for TA_S). ~~The increase in the time step length
434 after 10 s is the same, irrespective of the algorithm. For a smaller time, both truncation time
435 stepping strategies differ for the estimate of the first time step only. The scheme using the
436 saturation based time stepping is penalized by the poor estimate of the first maximum allowed
437 saturation change. This leads to the estimate of the first time step magnitude that was too long
438 for reaching convergence. The time step evolution is very similar for the three strategies: a
439 linear increase until around 0.1s, followed by a very slow increase until 20-30s and a regular
440 increase until the end of the simulation. ST_Δψ and TA_T strategies lead to the same time~~

441 steps when time reaches 1s. The time step sizes remain smaller for TA_S which explains the
442 significant higher number of iterations required to solve this test case.

443

444 *TC3: Infiltration/evaporation in a heterogeneous soil*

445 This case study simulates infiltration in an initially dry heterogeneous soil with a succession
446 of rainfall and evaporations as upper boundary conditions during 35 days. This problem
447 differs from the two previous cases by the soil heterogeneity and also by the non-monotonic
448 boundary conditions at the soil surface. It is expected that non-monotonic discontinuous
449 boundary conditions will increase the difficulty of finding accurate solutions. The soil profile
450 consists of three 60 cm thick layers. The layers are discretized using cells with the size of 0.10

451 cm. The prescribed fluxes are changing every day. For a given time, these fluxes are linearly
452 interpolated. To avoid a too rough time discretization of these boundary conditions, the
453 maximum time step magnitude has been fixed at 0.20 day. ~~The maximum time step~~
454 magnitude is chosen as 0.20 days to avoid a too rough discretization of the upper boundary
455 conditions. The initial time step is set to 10^{-5} day.

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

458 The standard iterative scheme fails to converge within the maximum number of iterations
459 (10^7) when the tolerance is not sufficiently restrictive. The detailed analyses of the
460 computation showed that the time step size was quite large compared to the more restrictive
461 conditions until day 28.0 where the infiltration fluxes were equal to 1.50 cm/day and where
462 the conditions were near saturation due to the previous infiltration period. This led to a
463 decrease of the time step to close to the minimum value (10^{-8} s), causing the procedure to

464 stop. More restrictive conditions lead to smaller time steps from the beginning of the
465 simulation and a better approximation of the solutions during the entire simulation.

466 The iterative scheme coupled with the truncation based time step strategy showed a
467 surprisingly unstable behavior for $\tau_r=10^{-3}$. The scheme did not converge for
468 $\tau_r \in [0.96 \cdot 10^{-3}; 1.04 \cdot 10^{-3}]$. The results presented in Table 7 and Figure 8 are obtained for
469 $\tau_r = 0.90 \cdot 10^{-3}$. At this stage of our work, we were not able to provide a meaningful
470 explanation for this effect.

471 The time-adaptive algorithm with the saturation based time stepping scheme is the most
472 efficient for an L_2 -norm greater than 10^{-4} . For more accurate results, the iterative method with
473 the time stepping strategy using the truncation error must be preferred. The impact of the time
474 stepping strategy for these two algorithms is shown in Figure 9 for approximately the same
475 L_2 -norm ($2.051 \cdot 10^{-3}$ within 1283 iterations for TA_S and $1.517 \cdot 10^{-3}$ within 6504 iterations for
476 ST_Δψ). The time step changes is related to the boundary conditions variations as expected.
477 The strategy based on the saturation variation leads to a longer time step than the strategy
478 using the time truncation error. This difference can be quite important (see the simulation
479 between days 25 and 30). The consequences of this difference are a reduced number of
480 iterations but also a less accurate computation, irrespective of the error norm.

481

482 **6. Summary and conclusions**

483 The solution of RE is complex and very time consuming due to its highly non-linear
484 properties. Several algorithms have been tested for the mixed-form of Richards equation,
485 including time-adaptive methods. Based on the numerical examples that differ in their

486 parameters (level of non-linearity) and in their initial and boundary conditions, the
487 conclusions and recommendations are:

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

491 2. The different algorithms have different convergence rates (accuracy improvement of
492 the scheme as a function of the computational costs). Therefore, an algorithm can be
493 very efficient for a given accuracy and less efficient for another level of precision.

494 However, for these three test cases and in average, the best performance in terms of
495 efficiency was obtained using a stopping criterion based on truncation error with its
496 corresponding time step strategy (ST $\Delta\psi$). Similar results were obtained by Kavetski
497 et al. (2001) for the pressure-based RE and by Kavetski and Binning (2004) for the
498 moisture-based RE.

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

502 4. ~~The use of both criteria (ϵ_ψ , the maximum variable difference between two iterations,~~
503 ~~ϵ_t , the The time truncation error)~~ should be implemented in the numerical codes using
504 the standard iterative procedure. The use of ~~ϵ_ψ , the maximum variable difference~~
505 ~~between two successive iterations~~ only, which is ~~the case in many numerical~~
506 ~~codes usually implemented~~, does not provide any information about the accuracy of the
507 time derivative approximation.

508 ~~5.~~ Our 1-dimensional examples showed that time-adaptive algorithm TA_T is very
509 sensitive to the type of problem to solve. The time-adaptive algorithm TA_S was less

510 efficient than the usual schemes. However, for a larger amount of elements like in 2D
511 or 3D problems, this conclusion might be different because the time dedicated to the
512 computation of the parameters can be significant higher, unless tabulated values are
513 used to evaluate the parameters and the required derivatives. ~~did not show a significant~~
514 advantage of the time-adaptive algorithm that avoids the computation of the
515 parameters for each iteration. However, this may depend on the number of elements
516 used for the spatial discretization, and this conclusion may be different for 2D or 3D
517 domains.

518

519 Depending on the type of the problem that must be solved (parameters behavior with respect
520 to the pressure, time variations of the boundary conditions), the time truncation errors may be
521 predominant compared to the error corresponding to the pressure changes between two
522 successive iterations. Therefore, we recommend ~~the use of both types of errors by~~
523 ~~implementing the truncation errors either in the convergence procedure (convergence reached~~
524 ~~if ϵ_v and ϵ_t are smaller than a user's defined tolerance)~~ the implementation of this stopping
525 criteria associated with ~~or in~~ the time stepping strategy as defined by Kavetski et al. (2001).

526

527

528 **References**

- 529 Bause, M., Knabner, P., 2004. Computation of variably saturated subsurface flow by adaptive
530 mixed hybrid finite element methods. *Adv. Water Resour.* 27, 565–581.
531 doi:10.1016/j.advwatres.2004.03.005
- 532 Belfort, B., Younes, A., Fahs, M., Lehmann, F., 2013. On equivalent hydraulic conductivity
533 for oscillation-free solutions of Richard's equation. *J. Hydrol.* 505, 202–217.
534 doi:10.1016/j.jhydrol.2013.09.047
- 535 Bergamaschi, L., Putti, M., 1999. Mixed finite elements and Newton-type linearizations for
536 the solution of Richards' equation. *Int. J. Numer. Methods Eng.* 45, 1025–1046.
537 doi:10.1002/(SICI)1097-0207(19990720)45
- 538 Bouchemella, S., Seridi, A., Alimi-Ichola, I., 2015. Numerical simulation of water flow in
539 unsaturated soils: comparative study of different forms of Richards's equation. *Eur. J.*
540 *Environ. Civ. Eng.* 19, 1–26. doi:10.1080/19648189.2014.926294
- 541 Celia, M.A., Bouloutas, E.T., Zarba, R.L., 1990. A general mass-conservative numerical
542 solution for the unsaturated flow equation. *Water Resour. Res.* 26, 1483–1496.
543 doi:10.1029/WR026i007p01483
- 544 [Clark, M. P., Fan Y., Lawrence D. M., Adam J. C., Bolster D., Gochis D. J., Hooper R. P.,](#)
545 [Kumar M., Leung L. R., Mackay D. S., Maxwell R. M., Shen C., Swenson S. C., Zeng, X.,](#)
546 [2015. Improving the representation of hydrologic processes in Earth System Models. *Water*](#)
547 [Resour. Res., 51, 5929–5956, doi:10.1002/2015WR017096.](#)
- 548 Crevoisier, D., Chanzy, A., Voltz, M., 2009. Evaluation of the Ross fast solution of Richards'
549 equation in unfavourable conditions for standard finite element methods. *Adv. Water Resour.*
550 32, 936–947. doi:10.1016/j.advwatres.2009.03.008
- 551 Diersch, H.-J.G., Perrochet, P., 1999. On the primary variable switching technique for
552 simulating unsaturated-saturated flows. *Adv. Water Resour.* 23, 271–301.
553 doi:10.1016/S0309-1708(98)00057-8
- 554 El Kadi, A.I., Ling, G., 1993. The Courant and Peclet Number criteria for the numerical
555 solution of the Richards Equation. *Water Resour. Res.* 29. doi:10.1029/93WR00929
- 556 Fahs, M., Younes, A., Lehmann, F., 2009. An easy and efficient combination of the Mixed
557 Finite Element Method and the Method of Lines for the resolution of Richards' Equation.
558 *Environ. Model. Softw.* 24, 1122–1126. doi:10.1016/j.envsoft.2009.02.010
- 559 Farthing, M.W., Kees, C.E., Miller, C.T., 2003. Mixed finite element methods and higher
560 order temporal approximations for variably saturated groundwater flow. *Adv. Water Resour.*
561 26, 373–394. doi:10.1016/S0309-1708(02)00187-2
- 562 Feddes, R.A., Kabat, P., Van Bakel, P.J.T., Bronswijk, J.J.B., Halbertsma, J., 1988. Modelling
563 soil water dynamics in the unsaturated zone — State of the art. *J. Hydrol.* 100, 69–111.
564 doi:10.1016/0022-1694(88)90182-5
- 565 Forsyth, P.A., Wu, Y.S., Pruess, K., 1995. Robust numerical methods for saturated-
566 unsaturated flow with dry initial conditions in heterogeneous media. *Adv. Water Resour.* 18,
567 25–38. doi:10.1016/0309-1708(95)00020-J
- 568 Gottardi, G., Venutelli, M., 2001. UPF: two-dimensional finite-element groundwater flow
569 model for saturated-unsaturated soils. *Comput. Geosci.* 27, 179–189. doi:10.1016/S0098-
570 3004(00)00082-0

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

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

577 Kavetski, D., Binning, P., 2002a. Adaptive backward Euler time stepping with truncation
578 error control for numerical modelling of unsaturated fluid flow. *Int. J. Numer. Methods Eng.*
579 53, 1301–1322. doi:10.1002/nme.329

580 Kavetski, D., Binning, P., 2002b. Noniterative time stepping schemes with adaptive
581 truncation error control for the solution of Richards equation. *Water Resour. Res.* 38.
582 doi:10.1029/2001WR000720

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

586 Kirkland, M.R., Hills, R.G., Wierenga, P.J., 1992. Algorithms for solving Richards’ equation
587 for variably saturated soils. *Water Resour. Res.* 28, 2049–2058.

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

591 Matthews, C.J., Braddock, R.D., Sander, G.C., 2004. Modeling flow through a one-
592 dimensional multi-layered soil profile using the Method of Lines. *Environ. Model. Assess.* 9,
593 103–113.

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

597 Miller, C.T., Williams, G.A., Kelley, C.T., Tocci, M.D., 1998. Robust solution of Richards’
598 equation for nonuniform porous media. *Water Resour. Res.* 34, 2599–2610.
599 doi:10.1029/98WR01673

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

602 Paniconi, C., Aldama, A.A., Wood, E.F., 1991. Numerical evaluation of iterative and
603 noniterative methods for the solution of the nonlinear Richards equation. *Water Resour. Res.*
604 27, 1147–1163.

605 Paniconi, C., Putti, M., 1994. A comparison of Picard and Newton iteration in the numerical
606 solution of multidimensional variably saturated flow problems. *Water Resour. Res.* 30, 3357–
607 3374. doi:10.1029/94WR02046

608 R. G. Hills, I.P., 1989. Modeling one-dimensional infiltration into very dry soils: 1. Model
609 development and evaluation. *Water Resour. Res.* 25. doi:10.1029/WR025i006p01259

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

612 Richards, L.A., 1931. Capillary conduction of liquids through porous medium. *J. Appl. Phys.*
613 1, 318–333. doi:10.1063/1.1745010

- 614 Romano, N., Brunone, B., Santini, A., 1998. Numerical analysis of one-dimensional
615 unsaturated flow in layered soils. *Adv. Water Resour.* 21, 315–324. doi:10.1016/S0309-
616 1708(96)00059-0
- 617 Ross, P.J., 2003. Modeling Soil Water and Solute Transport—Fast, Simplified Numerical
618 Solutions. *Agron. J.* 95, 1352. doi:10.2134/agronj2003.1352
- 619 Ross, P.J., Bristow, K.L., 1990. Simulating Water Movement in Layered and Gradational
620 Soils Using the Kirchhoff Transform. *Soil Sci. Soc. Am. J.* 54, 1519.
621 doi:10.2136/sssaj1990.03615995005400060002x
- 622 Thomas, R.M., Gladwell, I., 1988. Variable-order variable-step algorithms for second-order
623 systems. Part I: The methods. *Int. J. Numer. Methods Eng.* 26, 39–53.
- 624 Tocci, M.D., Kelley, C.T., Miller, C.T., 1997. Accurate and economical solution of the
625 pressure-head form of Richards' equation by the method of lines. *Adv. Water Resour.* 20, 1–
626 14. doi:10.1016/S0309-1708(96)00008-5
- 627 van Dam, J.C., Feddes, R.A., 2000. Numerical simulation of infiltration, evaporation and
628 shallow groundwater levels with the Richards equation. *J. Hydrol.* 233, 72–85.
629 doi:10.1016/S0022-1694(00)00227-4
- 630 van Genuchten, M.T., 1980. A Closed-form Equation for Predicting the Hydraulic
631 Conductivity of Unsaturated Soils¹. *Soil Sci. Soc. Am. J.* 44, 892.
632 doi:10.2136/sssaj1980.03615995004400050002x
- 633 Varado, N., Braud, I., Ross, P.J., 2006a. Development and assessment of an efficient vadose
634 zone module solving the 1D Richards' equation and including root extraction by plants. *J.*
635 *Hydrol.* 323, 258–275. doi:10.1016/j.jhydrol.2005.09.015
- 636 Varado, N., Braud, I., Ross, P.J., Haverkamp, R., 2006b. Assessment of an efficient numerical
637 solution of the 1D Richards' equation on bare soil. *J. Hydrol.* 323, 244–257.
638 doi:10.1016/j.jhydrol.2005.07.052
- 639 [Vergnes, J. P., Decharme, B., Alkama, R., Martin, E., Habets, F., Douville, H., 2012. A](#)
640 [simple groundwater scheme for hydrological and climate applications: Description and offline](#)
641 [evaluation over France. *Journal of Hydrometeorology*, 13\(4\), 1149-1171.](#)
- 642 Williams, G.A., Miller, C.T., Kelley, C.T., 2000. Transformation approaches for simulating
643 flow in variably saturated porous media. *Water Resour. Res.* 36, 923–934.
644 doi:10.1029/1999WR900349
- 645 Zha, Y., Shi, L., Ye, M., Yang, J., 2013. A generalized Ross method for two- and three-
646 dimensional variably saturated flow. *Adv. Water Resour.* 54, 67–77.
647 doi:10.1016/j.advwatres.2013.01.002
- 648 Zha, Y., Tso, M.C.-H., Shi, L., Yang, J., 2015. Comparison of Non iterative Algorithms
649 Based on Different Forms of Richards' Equation. *Environ. Model. Assess.* 21, 357–370.
650 doi:10.1007/s10666-015-9467-1

651

652

653 Section d'équation 1

Code de champ modifié

Code de champ modifié

654
655

APPENDIX 1.

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

658

$$\begin{cases} \frac{\theta_i^{n+1} - \theta_i^n}{\Delta t} + S_w S_0 \frac{\psi_i^{n+1} - \psi_i^n}{\Delta t} + \frac{q_{i+}^{n+1} - q_{i-}^{n+1}}{\Delta z_i} = f_i \\ q_{i-}^{n+1} = -K_{i-} \left(\frac{\psi_i^{n+1} - \psi_{i-1}^{n+1}}{\Delta z_{i-}} - 1 \right) \\ q_{i+}^{n+1} = -K_{i+} \left(\frac{\psi_{i+1}^{n+1} - \psi_i^{n+1}}{\Delta z_{i+}} - 1 \right) \end{cases} \quad (A1)$$

Code de champ modifié

660

661 where n is the time step, K_{i-} is the inter-block conductivity between cell i and $(i-1)$ defined by

662 $K_{i-} = \frac{\Delta z_{i-1} K(\psi_{i-1}) + \Delta z_i K(\psi_i)}{\Delta z_{i-1} + \Delta z_i}$, K_{i+} is the inter-block conductivity between cell i and $(i+1)$

663 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

664 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
665 $(i+1)$.

666

667

668 The residual is:

669

$$670 \quad R(\psi_i^{n+1,k}) = \Delta z_i (\theta_i^{n+1,k} - \theta_i^n) + \Delta z_i S_w S_0 (\psi_i^{n+1,k} - \psi_i^n) + \Delta t (q_{i+}^{n+1,k} - q_{i-}^{n+1,k}) - \Delta t \Delta z_i f_i \quad (A2)(A1)$$

Code de champ modifié

671

672 where k is the iteration counter.

673

674 The residual derivatives are:

675

$$\begin{aligned} \frac{\partial R(\psi_i^{n+1,k})}{\partial \psi_{i-1}^{n+1,k}} &= -\Delta t \frac{\partial q_{i-}^{n+1,k}}{\partial \psi_{i-1}^{n+1,k}} \\ \frac{\partial R(\psi_i^{n+1,k})}{\partial \psi_i^{n+1,k}} &= \Delta z_i \frac{d\theta_i^{n+1,k}}{d\psi_i^{n+1,k}} + \Delta z_i S_w S_0 + \Delta t \left(\frac{\partial q_{i+}^{n+1,k}}{\partial \psi_i^{n+1,k}} - \frac{\partial q_{i-}^{n+1,k}}{\partial \psi_i^{n+1,k}} \right) \\ \frac{\partial R(\psi_i^{n+1,k})}{\partial \psi_{i+1}^{n+1,k}} &= \Delta t \frac{\partial q_{i+}^{n+1,k}}{\partial \psi_{i+1}^{n+1,k}} \end{aligned} \quad (A3)(A1)$$

Code de champ modifié

677

678 Therefore, the system to solve is:

679

$$-\Delta t \frac{\partial q_{i-}^{n+1,k}}{\partial \psi_{i-}^{n+1,k}} \Delta \psi_{i-}^{n+1,k+1} +$$

680

$$\left[\Delta z_i \frac{d\theta_i^{n+1,k}}{d\psi_i^{n+1,k}} + \Delta z_i S_w S_0 + \Delta t \left(\frac{\partial q_{i+}^{n+1,k}}{\partial \psi_i^{n+1,k}} - \frac{\partial q_{i-}^{n+1,k}}{\partial \psi_i^{n+1,k}} \right) \right] \Delta \psi_i^{n+1,k+1} +$$

$$\Delta t \frac{\partial q_{i+}^{n+1,k}}{\partial \psi_{i+1}^{n+1,k}} \Delta \psi_{i+1}^{n+1,k+1} =$$

$$-\Delta z_i (\theta_i^{n+1,k} - \theta_i^n) - \Delta z_i S_w S_0 (\psi_i^{n+1,k} - \psi_i^n) - \Delta t (q_{i+}^{n+1,k} - q_{i-}^{n+1,k}) + \Delta t \Delta z_i f_i$$

681

(A4)(A1)

Code de champ modifié

682

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

684

685

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

(A5)(A1)

Code de champ modifié

686

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

688

689

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

(A6)(A1)

Code de champ modifié

690

691 The component of the vector of the residuals \mathbf{R} is given by equation (A2) and the692 coefficients of the matrix \mathbf{R}' for cell i are:

693

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

694

$$R'_{i,i} = \Delta z_i \left[\frac{d\theta_i^{n+1,k}}{d\psi_i^{n+1,k}} + S_w S_0 \right] - \Delta t \left[\frac{\partial K_{i+}^{n+1,k}}{\partial \psi_i^{n+1,k}} \left(\frac{\psi_{i+1}^{n+1,k} - \psi_i^{n+1,k}}{\Delta z_{i+}} - 1 \right) - \frac{K_{i+}^{n+1,k}}{\Delta z_{i+}} \right]$$

(A7)(A1)

Code de champ modifié

$$+ \Delta t \left[\frac{\partial K_{i-}^{n+1,k}}{\partial \psi_i^{n+1,k}} \left(\frac{\psi_i^{n+1,k} - \psi_{i-1}^{n+1,k}}{\Delta z_{i-}} - 1 \right) + \frac{K_{i-}^{n+1,k}}{\Delta z_{i-}} \right]$$

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

695

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

697

$$R_1(\psi_1^{n+1,k}) = \Delta z_1 \left[(\theta_1^{n+1,k} - \theta_1^n) + S_w S_0 (\psi_1^{n+1,k} - \psi_1^n) \right] + \Delta t (q_{1+}^{n+1} - q_{BC}) - \Delta t \Delta z_1 f_1 \quad (A8)(A1)$$

Code de champ modifié

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

Code de champ modifié

Code de champ modifié

$$R'_{1,1} = \Delta z_1 \left(\frac{d\theta_1^{n+1,k}}{d\psi_1^{n+1,k}} + S_w S_0 \right) - \Delta t \left[\frac{\partial K_{1+}^{n+1,k}}{\partial \psi_1^{n+1,k}} \left(\frac{\psi_2^{n+1,k} - \psi_1^{n+1,k}}{\Delta z_{1+}} - 1 \right) - \frac{K_{1+}^{n+1,k}}{\Delta z_{1+}} \right]$$

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

(A9)(A1)

Code de champ modifié

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

$$R_N = \Delta z_N \left[(\theta_N^{n+1,k} - \theta_N^n) + S_w S_0 (\psi_N^{n+1,k} - \psi_N^n) \right] + \Delta t (q_{BC} - q_{N-}^{n+1,k}) - \Delta t \Delta z_N f_N \quad (A10)(A1)$$

(A10)(A1)

Code de champ modifié

and its derivatives

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

$$R'_{N,N} = \Delta z_N \left(\frac{d\theta_N^{n+1,k}}{d\psi_N^{n+1,k}} + S_w S_0 \right) + \Delta t \left[\frac{\partial K_{N-}^{n+1,k}}{\partial \psi_N^{n+1,k}} \left(\frac{\psi_N^{n+1,k} - \psi_{N-1}^{n+1,k}}{\Delta z_{N-}} - 1 \right) + \frac{K_{N-}^{n+1,k}}{\Delta z_{N-}} \right]$$

(A11)(A1)

Code de champ modifié

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

$$q_{1-}^{n+1,k} = -K_{1-} \left(\frac{\psi_1^{n+1,k} - \psi_{BC}}{\Delta z_1 / 2} - 1 \right) \quad (A12)(A1)$$

(A12)(A1)

Code de champ modifié

And the derivative is:

$$\frac{\partial q_{1-}^{n+1,k}}{\partial \psi_1^{n+1,k}} = -\frac{\partial K_{1-}^{n+1,k}}{\partial \psi_1^{n+1,k}} \left(\frac{\psi_1^{n+1,k} - \psi_{BC}}{\Delta z_1 / 2} - 1 \right) - \frac{K_{1-}^{n+1,k}}{\Delta z_1 / 2} \quad (A13)(A1)$$

(A13)(A1)

Code de champ modifié

The corresponding residual and the matrix coefficients are:

722 $R_1 = \Delta z_1 \left[(\theta_1^{n+1,k} - \theta_1^n) + S_w s_0 (\psi_1^{n+1,k} - \psi_1^n) \right] + \Delta t (q_{1+}^{n+1,k} - q_{1-}^{n+1,k}) - \Delta t \Delta z_1 f_1$

723 and

724
$$R'_{1,1} = \Delta z_1 \left(\frac{d\theta_1^{n+1,k}}{d\psi_1^{n+1,k}} + S_w s_0 \right) - \Delta t \left[\frac{\partial K_{1+}^{n+1,k}}{\partial \psi_1^{n+1,k}} \left(\frac{\psi_2^{n+1,k} - \psi_1^{n+1,k}}{\Delta z_{1+}} - 1 \right) - \frac{K_{1+}^{n+1,k}}{\Delta z_{1+}} \right]$$

$$+ \Delta t \left[\frac{\partial K_{1-}^{n+1,k}}{\partial \psi_1^{n+1,k}} \left(\frac{\psi_1^{n+1,k} - \psi_{BC}}{\Delta z_1 / 2} - 1 \right) + \frac{K_{1-}^{n+1,k}}{\Delta z_1 / 2} \right]$$

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

725

726 Similarly, if the pressure is prescribed at the soils column's bottom, we have:

727

728 $R_N = \Delta z_N \left[(\theta_N^{n+1,k} - \theta_N^n) + S_w s_0 (\psi_N^{n+1,k} - \psi_N^n) \right] + \Delta t (q_{N+}^{n+1,k} - q_{N-}^{n+1,k}) - \Delta t \Delta z_N f_N$

729 and

730
$$R'_{N-1,N} = \Delta t \left[\frac{\partial K_{N-}^{n+1,k}}{\partial \psi_{N-1}^{n+1,k}} \left(\frac{\psi_N^{n+1,k} - \psi_{N-1}^{n+1,k}}{\Delta z_{N-}} - 1 \right) - \frac{K_{N-}^{n+1,k}}{\Delta z_{N-}} \right]$$

$$R'_{N,N} = \Delta z_N \left(\frac{d\theta_N^{n+1,k}}{d\psi_N^{n+1,k}} + S_w s_0 \right) - \Delta t \left[\frac{\partial K_{N+}^{n+1,k}}{\partial \psi_N^{n+1,k}} \left(\frac{\psi_{BC} - \psi_N^{n+1,k}}{\Delta z_N / 2} - 1 \right) - \Delta t \frac{K_{N+}^{n+1,k}}{\Delta z_N / 2} \right]$$

$$+ \Delta t \left[\frac{\partial K_{N-}^{n+1,k}}{\partial \psi_N^{n+1,k}} \left(\frac{\psi_N^{n+1,k} - \psi_{N-1}^{n+1,k}}{\Delta z_{N-}} - 1 \right) + \frac{K_{N-}^{n+1,k}}{\Delta z_{N-}} \right]$$

731

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

(A14)(A1)

Code de champ modifié

(A15)(A1)

Code de champ modifié

(A16)(A1)

Code de champ modifié

(A17)(A1)

Code de champ modifié

List of Tables

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

Table 2: Domain size (L), initial conditions (IC), boundary conditions at the soil surface (BC₀) and at the soil bottom (BC₁), saturated hydraulic conductivity (K_s), residual and saturated water contents (θ_r, θ_s) and shape parameters (α, n) for the different test cases. Length and time units are centimeters and seconds.

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

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

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).

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

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} , $\tau_r=0.90 \cdot 10^{-3}$).

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

	Standard iterative algorithm					Time-adaptive algorithm	
	Heuristic	Time stepping		Stopping criterion		Truncation	Saturation
		Truncation	Saturation	Pressure	Truncation		
	(19)(19)	(25)(25) (26)	(20)(20) (21)(21)	(27)(27)	(28)(28)	(25)(25) (26)	(20)(20) (21)(21)
SH_Δψ	x			x			
SH_Δψ_Δt	x			x	x		
ST_Δψ		x		x			
SS_Δψ_Δt			x	x	x		
TA_T						x	
TA_S							x

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

- Mis en forme : Français (France)

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

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.

Tol.	Algorithm	L₁	L₂	L_∞	N_{trunc}	N_{sol}
10 ⁻⁵	SH_Δψ	1.918 10 ⁻³	8.829 10 ⁻³	0.106		2177
	SH_Δψ_Δt	8.391 10 ⁻⁶	6.459 10 ⁻⁵	8.782 10 ⁻⁴	542371	615880
	ST_Δψ	3.968 10 ⁻⁴	1.045 10 ⁻³	3.512 10 ⁻³		6160
	SS_Δψ_Δt	1.136 10 ⁻⁵	3.406 10 ⁻⁵	2.817 10 ⁻⁴	252	3920446
10 ⁻⁴	SH_Δψ	2.557 10 ⁻³	1.375 10 ⁻²	0.168		1701
	SH_Δψ_Δt	7.818 10 ⁻⁵	2.259 10 ⁻⁴	1.593 10 ⁻³	170438	194420
	ST_Δψ	1.331 10 ⁻³	1.316 10 ⁻³	1.181 10 ⁻²		1950
	SS_Δψ_Δt	8.607 10 ⁻⁶	3.525 10 ⁻⁵	3.899 10 ⁻⁴	154597	392041
10 ⁻³	SH_Δψ	3.956 10 ⁻³	1.166 10 ⁻²	0.125		1312
	SH_Δψ_Δt	2.320 10 ⁻⁴	7.553 10 ⁻⁴	7.883 10 ⁻³	52723	60303
	ST_Δψ	2.241 10 ⁻³	5.702 10 ⁻³	1.792 10 ⁻²		620
	SS_Δψ_Δt	6.567 10 ⁻⁵	1.585 10 ⁻⁴	1.453 10 ⁻³	9895	39110
10 ⁻²	SH_Δψ	6.559 10 ⁻³	1.716 10 ⁻²	0.119		1018
	SH_Δψ_Δt	2.224 10 ⁻³	7.923 10 ⁻³	7.111 10 ⁻²	15540	17888
	ST_Δψ	9.954 10 ⁻³	2.630 10 ⁻²	8.727 10 ⁻²		243
	SS_Δψ_Δt	8.283 10 ⁻⁴	2.271 10 ⁻³	1.478 10 ⁻²	862	3804

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

Tol.	Algorithm	L₁	L₂	L_∞	N_{param}	N_{sol}
10 ⁻⁵	TA_T	5.016 10 ⁻³	2.376 10 ⁻²	0.269	32197	35938
	TA_S	6.152 10 ⁻⁶	2.429 10 ⁻⁵	2.561 10 ⁻⁴	9316700	9322946
10 ⁻⁴	TA_T	5.598 10 ⁻³	2.580 10 ⁻²	0.284	10169	11520
	TA_S	2.839 10 ⁻⁵	1.363 10 ⁻⁴	1.654 10 ⁻³	931616	938144
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	93114	100898
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	18292

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

Tol.	Algorithm	L₁	L₂	L_∞	N_{trunc}	N_{sol}
10 ⁻⁵	SH_Δψ	6.966 10 ⁻³	1.818 10 ⁻²	5.878 10 ⁻²		573
	SH_Δψ_Δt	3.697 10 ⁻⁴	9.766 10 ⁻⁴	3.332 10 ⁻³	53769	59643
	ST_Δψ	1.578 10 ⁻⁴	4.254 10 ⁻⁴	2.451 10 ⁻³		3503
	SS_Δψ_Δt	-	-	-	-	n. c.
10 ⁻⁴	SH_Δψ	6.966 10 ⁻³	1.818 10 ⁻²	5.878 10 ⁻²		509
	SH_Δψ_Δt	6.968 10 ⁻⁴	1.979 10 ⁻³	5.726 10 ⁻³	16557	18428
	ST_Δψ	5.814 10 ⁻⁴	1.492 10 ⁻³	6.711 10 ⁻³		1033
	SS_Δψ_Δt	3.279 10 ⁻⁶	1.239 10 ⁻⁵	8.603 10 ⁻⁵	0	2474120
10 ⁻³	SH_Δψ	6.966 10 ⁻³	1.818 10 ⁻²	5.878 10 ⁻²		410
	SH_Δψ_Δt	3.699 10 ⁻³	9.761 10 ⁻³	3.275 10 ⁻²	4830	5444
	ST_Δψ	1.553 10 ⁻³	4.226 10 ⁻³	2.457 10 ⁻²		317
	SS_Δψ_Δt	2.355 10 ⁻⁵	6.230 10 ⁻⁵	2.341 10 ⁻⁴	0	247426
10 ⁻²	SH_Δψ	6.892 10 ⁻³	1.800 10 ⁻²	5.780 10 ⁻²		309
	SH_Δψ_Δt	9.135 10 ⁻³	2.409 10 ⁻²	7.925 10 ⁻²	376	580
	ST_Δψ	2.756 10 ⁻³	1.134 10 ⁻²	7.715 10 ⁻²		180
	SS_Δψ_Δt	2.973 10 ⁻⁴	7.884 10 ⁻⁴	3.252 10 ⁻³	0	24757

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 10⁷ iterations).

Tol.	Algorithm	L₁	L₂	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	1136193	1136199
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	113616	113616
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	11358	11358
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

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

Tol.	Algorithm	L₁	L₂	L_∞	N_{trunc}	N_{sol}
10 ⁻⁵	SH_Δψ	9.994 10 ⁻³	1.119 10 ⁻²	1.554 10 ⁻²		1644
	SH_Δψ_Δt	6.612 10 ⁻⁴	7.346 10 ⁻⁴	1.116 10 ⁻³	171636	190588
	ST_Δψ	6.830 10 ⁻⁴	7.775 10 ⁻⁴	1.648 10 ⁻³		16984
	SS_Δψ_Δt	7.185 10 ⁻⁵	7.935 10 ⁻⁵	1.297 10 ⁻⁴	197481	1646346
10 ⁻⁴	SH_Δψ	6.664 10 ⁻³	7.280 10 ⁻³	1.033 10 ⁻²		1734
	SH_Δψ_Δt	3.512 10 ⁻³	3.898 10 ⁻³	5.811 10 ⁻³	57312	63956
	ST_Δψ	1.300 10 ⁻³	1.517 10 ⁻³	2.412 10 ⁻³		6504
	SS_Δψ_Δt	5.380 10 ⁻⁵	6.536 10 ⁻⁵	1.010 10 ⁻⁴	41073	186351
10 ⁻³	SH_Δψ	-	-	-		n.c.
	SH_Δψ_Δt	2.625 10 ⁻³	2.899 10 ⁻³	4.971 10 ⁻³	22047	24779
	ST_Δψ	4.730 10 ⁻³	5.422 10 ⁻³	1.036 10 ⁻²		1297*
	SS_Δψ_Δt	7.569 10 ⁻⁴	8.820 10 ⁻⁴	1.402 10 ⁻³	16474	31276
10 ⁻²	SH_Δψ	-	-	-		n.c.
	SH_Δψ_Δt	5.493 10 ⁻³	6.306 10 ⁻³	1.171 10 ⁻³	7438	8812
	ST_Δψ	6.621 10 ⁻³	7.402 10 ⁻³	1.042 10 ⁻²		810
	SS_Δψ_Δt	7.511 10 ⁻³	8.780 10 ⁻³	1.378 10 ⁻²	5838	7535

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 10⁷ iterations, * convergence failed for 10⁻³, $\tau_r=0.90 \cdot 10^{-3}$).

Tol.	Algorithm	L₁	L₂	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	1357075	1357160
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	135386	135498
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	13314	13397
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

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

List of Figures

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

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).

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

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

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

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

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

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).

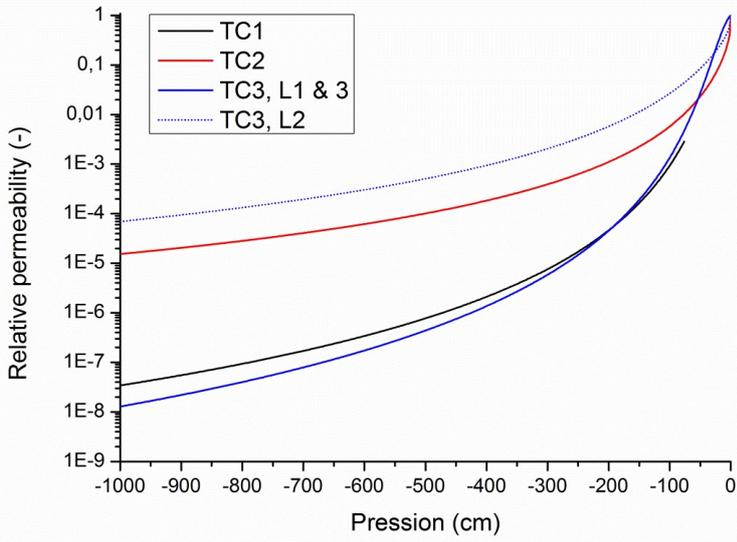


Figure 1: Relative permeability as a function of the pressure for the three test cases (L1, L2 and L3 are the three layers for test case 3).

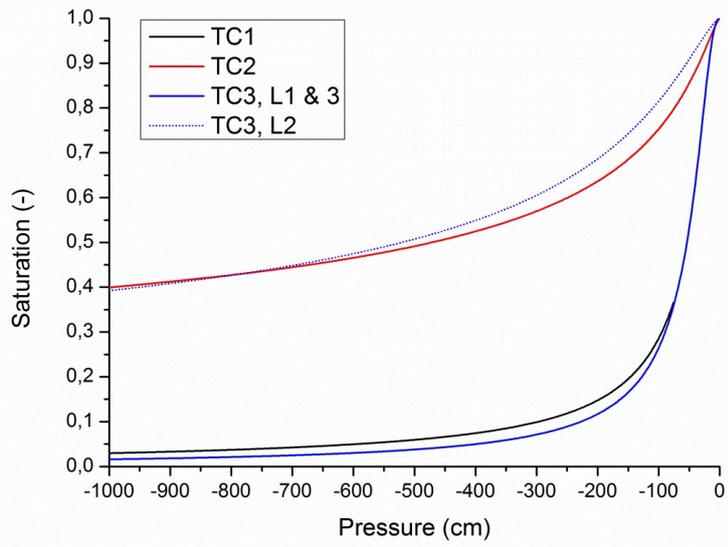


Figure 2: Water saturation as a function of the pressure for the three test cases (L1, L2 and L3 are the three layers for test case 3).

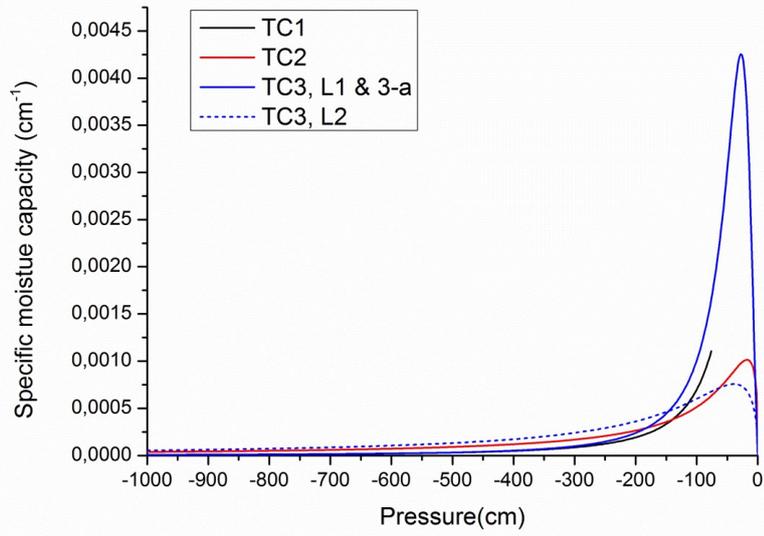


Figure 3: Specific moisture capacity as a function of the pressure for the three test cases (L1, L2 and L3 are the three layers for test case 3).

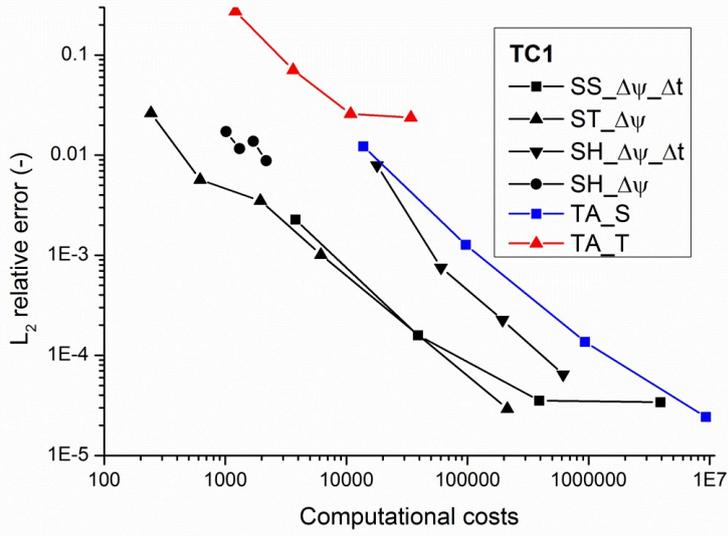


Figure 4: Evolution of the L_2 relative error with computational costs for TC1.

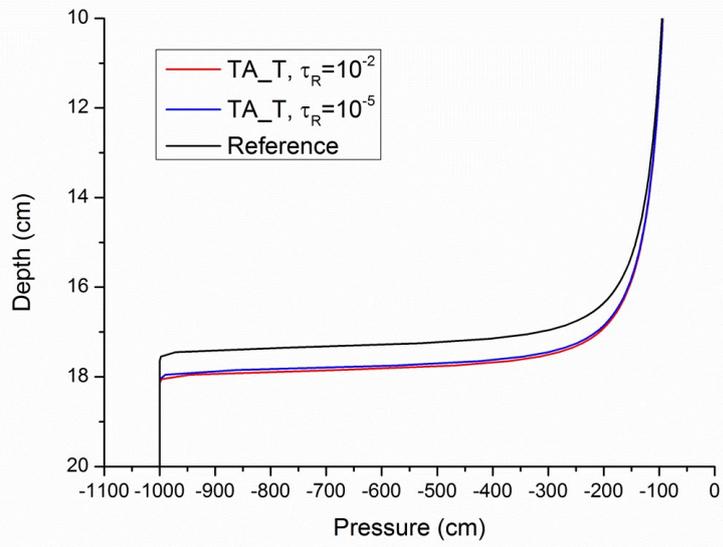


Figure 5: Pressure profiles in the domain for the TA_T algorithm.

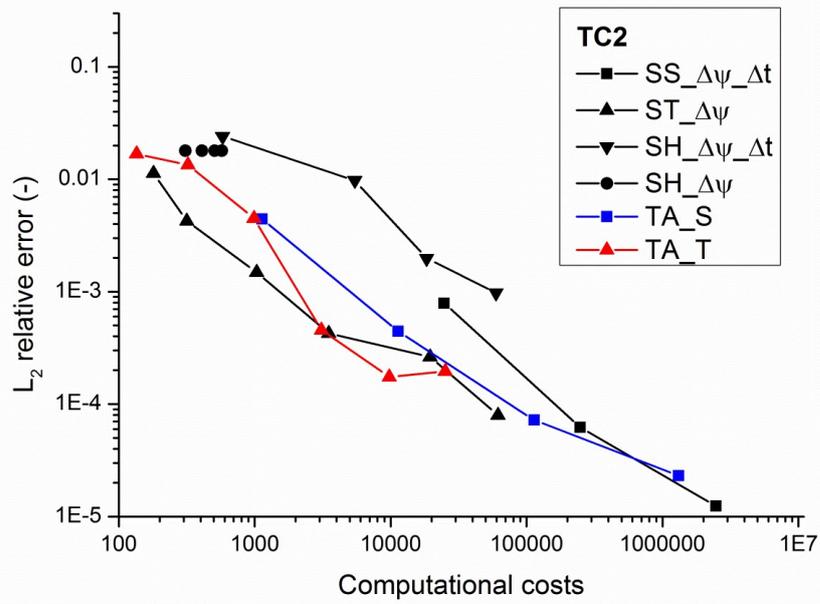


Figure 6: Evolution of the L_2 relative error with computational costs for TC2.

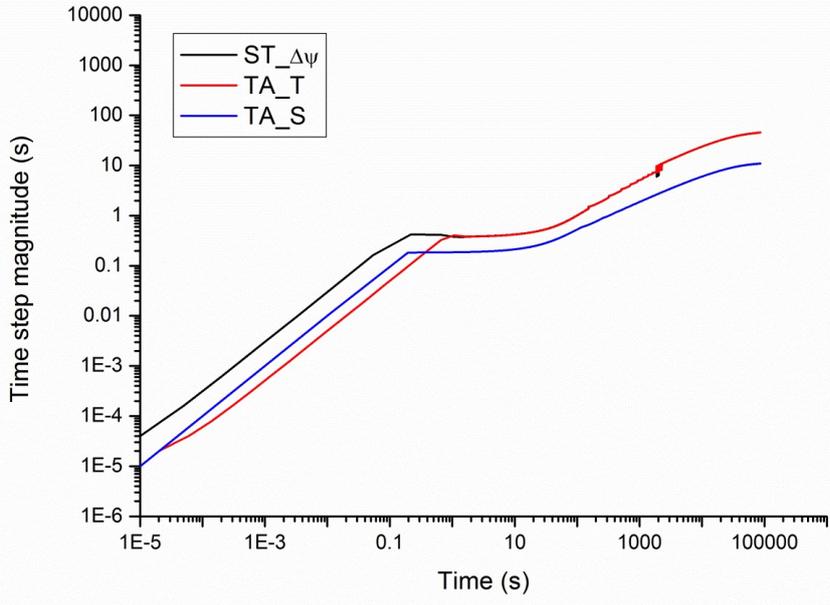
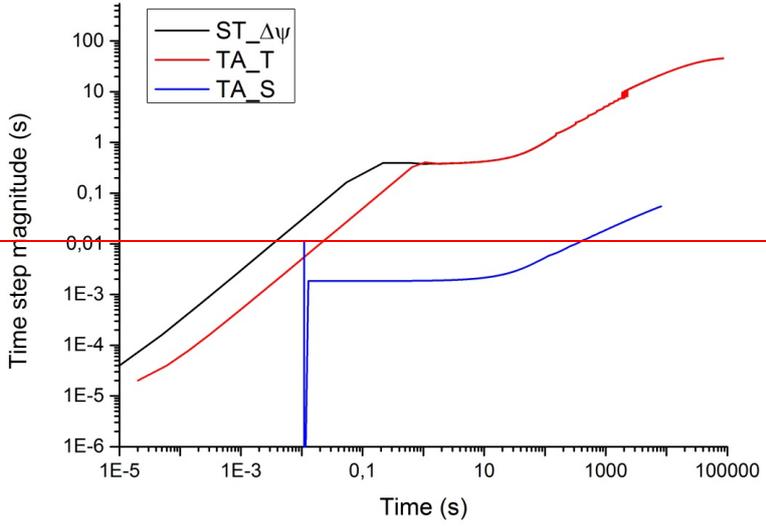


Figure 7: Time step magnitudes during the simulation for TC2.

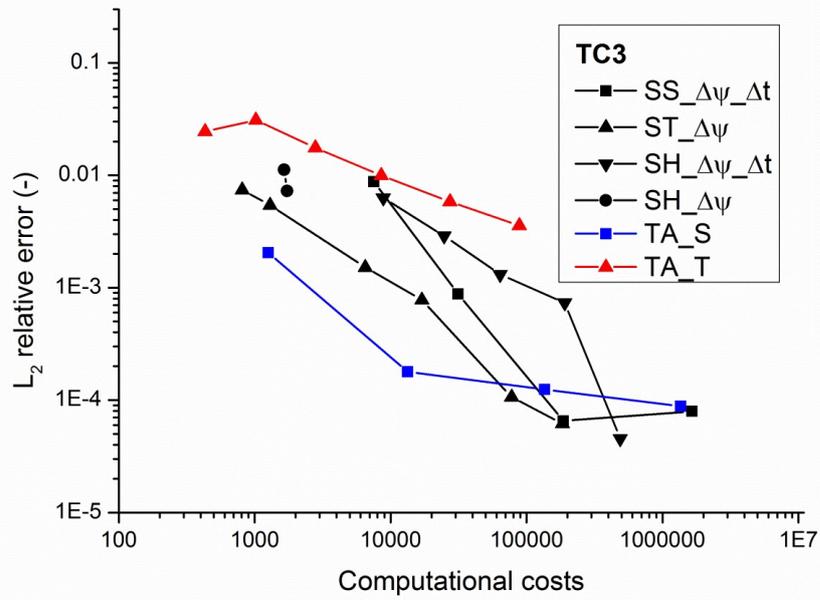


Figure 8: Evolution of the L_2 relative error with computational costs for TC3.

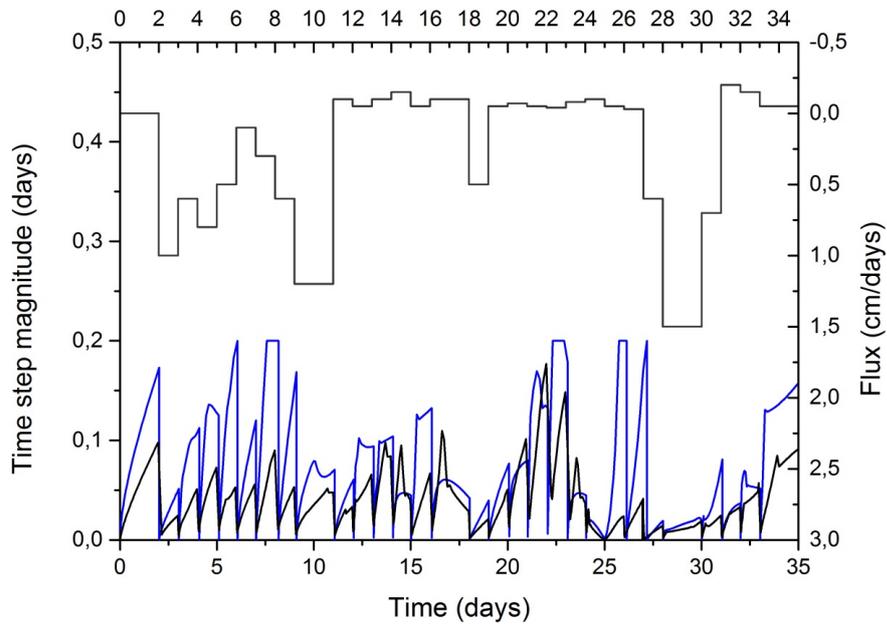


Figure 9: Time step magnitudes during the simulation for TC3 for the time stepping strategy based on truncation error (TA_S in blue, TA_T in black, time varying boundary conditions at the top).