COMPARISON OF FINITE DIFFERENCE SCHEMES FOR FLUID FLOW IN UNSATURATED POROUS MEDIUM (SOIL)

RAMESH CHANDRA TIMSINA[1] AND KEDAR NATH UPRETY[2]

[1] *Department of Mathematics, Patan Multiple Campus, Tribhuvan University, Kathmandu, Nepal*

[2] *Central Department of Mathematics, Tribhuvan University, Kathmandu, Nepal*

Abstract: Water movement in unsaturated porous medium (soil) can be expressed by Richards equation with the mass conservation law and Darcy–Buckingham's law. This equation can be expressed in three different forms as pressure head based, moisture content based and mixed from. In this study, we solve one dimensional Richards Equation in mixed form numerically using finite difference method with various time–stepping schemes: Forward Euler, Backward Euler, Crank–Nicolson and a Stabilized Runge–Kutta–Legendre Super Time-Stepping and we compare their performances using Dirichlet boundary condition on an isotropic homogeneous vertical soil column.

Keywords: Finite Difference Methods, Richards Equation, Kirchhoff Transformation, Super Time–Stepping Schemes, Infiltration.

1. INTRODUCTION

Water flow in unsaturated porous media (soil) is an important phenomena in groundwater hydrology. Water movement phenomena in unsaturated zone creates an emerging and realistic problems like contaminant transport, water–added transport of solutes and predicting water percolation in groundwater hydrology. In unsaturated zone the flow of water ascribable to capillary action and gravitational potential and is assumed to obey the classical Richards Equation [1]. Richards Equation is the combine from of mass conservation law and Darcy–Buckingham's law [2]. Consequently, this equation has three different forms as pressure head, moisture content and mixed from depending on either moisture content θ and pressure head ψ as the dependent variable. The constitutive (experimental) relationship between $\theta = \theta(z, t)$ and $\psi = \psi(z, t)$ allows the conversion from one another. As shown in [1] the couple or mixed form of Richards Equation takes the following form.

$$(1.1) \qquad \frac{\partial \theta}{\partial t} - \nabla . K(\psi) \nabla \psi - \frac{\partial K}{\partial z} = S(z, t),$$

where, θ is the volumetric moisture content, ψ is the pressure head $K(\psi)$ is the unsaturated hydraulic conductivity, describes the behavior of water flow that can move through pore

space, and depends on the permeability of the material used along with the properties of fluid [2]. $S(z,t)$ is the absorption or evapotranspiration rate for the root zone.

Constitutive relations between $\theta = \theta(z,t)$ and $\psi = \psi(z,t)$ and between K and ψ are developed appropriately, which consequently give nonlinear behavior to equation (1.1). To develop the accurate and reliable approximation of these relations are in general difficult and also a challenging task. Mostly, the choice of appropriate parameters are either from field measurements or laboratory experiments. But to gather these parameters, certainly it is an expensive task and the poorer condition is that these are limited to particular cases only. Moreover, the most widely used experimental constitutive relations for the moisture content and hydraulic conductivity are due to the work of Gardner, Brooks and Corry, van Genuchten and Haverkamp et al. [3]. Among these we use the following popular model from ground water hydrology due to Haverkamp et al. [4]

$$(1.2) \qquad \theta(\psi) = \theta_r + \frac{\alpha(\theta_s - \theta_r)}{\alpha + |\psi|^\beta}, K(\psi) = \frac{K_s A}{A + |\psi|^\gamma},$$

where θ_s and θ_r represent the saturated and residual moisture content respectively, K_s corresponds to the saturated hydraulic conductivity, and A, α, β, γ are dimensionless soil parameters.

The consecutive relations $K(\psi)$ and $\theta(\psi)$ in equation (1.2) have a property to change dramatically over a small range of ψ. Moreover, adopting these relations in Richards Equation (1.1), it becomes a highly nonlinear PDE in which analytical solution is rare and limited for particular case only. In groundwater regime, the movement of water in unsaturated soil is really a complicated process. Because of its complex nature, lack of absolute and reliable analytical solution to predict the flow movement via Richards Equation, numerous numerical methods have been developed. In recent years considerable amount of highly performable numerical methods are used to obtain the robust and reliable solution of the problems related to unsaturated flow. But there is a discrepancy of accuracy between the methods. Furthermore, they have assured a high computational cost and are inefficient for many realistic cases. Indeed, the expressions of equation (1.2) make the Richards Equation (1.1) highly nonlinear, it is customarily important to develop an efficient and accurate numerical approximation schemes for reliable solution. For this reason numerical approximation in infiltration problems are still considered to be one of the most important topics in groundwater hydrology.

Number of procedures developed as [5], [6], [7], [4], [8], [9] based on finite difference, finite element, finite volume, and Adaptive time–stepping strategies [10] for spatial discretization techniques to the partial differential equations have been used to approximate the numerical solution of Richards Equation (1.1).

Forward Euler scheme which is used to obtain the solution of Richards Equation (a parabolic PDE) has a strong restrictive stability criteria. With this reason researchers have not paid an effective cost to develop Forward Euler scheme for the Richards Equation. Most of the research work are devoted in developing Backward Euler scheme. Although Backward Euler schemes are unconditionally stable but with the iterative process in which the inverse matrix involved, they all turn out to be computationally high cost and in certain

circumstances unrealistic. Because of the presence of nonlinear behavior in the problem, to implement Forward Euler schemes, no mater which numerical approaches we use, the problem has to be linearized somehow at some stage. Indeed, some iterative approaches need to be applied to tackle this highly nonlinear problem to obtain the desired solution [3, 11].

Current trends in computational regime is to develop efficient parallelizable algorithm for High-performing Computers. To make the numerical procedures parallelizable, it is considered that the algorithm should be iteration free. For this, appropriate use of Forward Euler scheme is the best option. In [12], numerical solution of a linearized Richards equation is explained. In [8, 13], stability analysis of an Forward Euler scheme for the Richards Equation is derived.

In this work, to reduce the highly nonlinear Richards Equation to a functional non linear parabolic form, Kirchhoff integral transformation technique is used. The transformed equation is then solved numerically using Forward Euler, Backward Euler, Crank–Nicolson and a Stabilized Runge–Kutta–Legendre Super Time–Stepping scheme with Dirichlet boundary conditions respectively. Then their performances are compared accordingly.

The aim of this research work is to implement Super Time–Stepping strategies to stabilize an Forward Euler scheme [14] to the Kirchhoff transformed Richards Equation with dirichlet boundary condition which is relatively a new numerical approach. The work presented here, describes and verifies the implementation procedures and accuracy of a stabilized Runge–Kutta–Legendre Super Time–Stepping strategy (RKL) at Forward Euler finite difference scheme with Dirichlet boundary condition to simulate flow in unsaturated porous media (soil) in a vertical homogeneous soil column. The effectualness of using our scheme is that it is easy to implement, can be easily extended to problems in higher dimensions, and the most important property that it is unconditionally stable.

The paper is arranged in the following. Section 2 briefly describes the Kirchhoff transformation to linearize the Richard Equation. In section 3, numerical methods based on finite difference schemes are presented. In section 4, different test cases are solved and the results and conclusions are in section 5.

2. Simplified One–Dimensional Richards Equation

Considering the one space dimension, the Richards Equation (1.1) without sink and source term becomes

$$(2.1) \qquad \frac{\partial \theta}{\partial t} = \frac{\partial}{\partial z}\left(K(\psi)\frac{\partial \psi}{\partial z}\right) - \frac{\partial K}{\partial z}.$$

Then the equation(2.1) is generally used to simulate infiltration experiments. These experiments begin with wetting soil on top of the ground surface, showing a clear connection with the Darcy–Buckingham's law. If the soil is dry, it should be powered water on the top of ground surface. We make the assumption that the infiltration begins with known water

pressure head at the top and bottom of the soil column. For this we use the following initial and boundary conditions.

$$(2.2) \quad \begin{cases} \psi(z, 0) = \psi_0(z), \ 0 < z < L, \\ \psi(0, t) = \beta_1(t), \ t > 0, \\ \psi(L, t) = \beta_2(t), \ t > 0. \end{cases}$$

2.1. **Kirchhoff Integral Transform.** Kirchhoff integral transformation transformed equation (2.1), with $h = \psi - z$ by defining

$$(2.3) \quad \phi(h) = \int_0^h \bar{K}(\lambda) d\lambda.$$

Since $K(h) > 0$ from (1.2), the function $\phi(h)$ is strictly increasing with $\bar{K}(h) = K(\psi)$. Differentiating both sides to equation (2.3), we obtain :

$$(2.4) \quad \frac{\partial \phi}{\partial z} = \frac{\partial \phi}{\partial h} \frac{\partial h}{\partial z} = \bar{K}(h) \frac{\partial(\psi - z)}{\partial z} = K(\psi)(\frac{\partial \psi}{\partial z} - 1) = K(\psi) \frac{\partial \psi}{\partial z} - K(\psi).$$

Again differentiating of equation (2.4),

$$(2.5) \quad \frac{\partial^2 \phi}{\partial z^2} = \frac{\partial(K(\psi) \frac{\partial \psi}{\partial z})}{\partial z} - \frac{\partial}{\partial z}(K(\psi)).$$

Using the equation (2.5), the Richards Equation (2.1) takes the form

$$(2.6) \quad \frac{\partial \bar{\theta}}{\partial t} = \frac{\partial^2 \phi}{\partial z^2}.$$

with $\bar{\theta}(\phi) = \theta(h)$. The corresponding initial and boundary conditions to the transformed equation (2.6) takes the following form

$$(2.7) \quad \begin{cases} \phi(z, 0) = \phi_0(z), \ 0 < z < L, \\ \phi(0, t) = \bar{\beta}_1(t), \ t > 0, \\ \phi(L, t) = \bar{\beta}_2(t), \ t > 0. \end{cases}$$

The Kirchhoff transformation transformed the nonlinear equation (2.1) to a linear parabolic problem (2.6). Also we note that it preserves the uniqueness result for the resulting problem [13].

3. NUMERICAL METHOD

To solve equation (2.6) numerically with the prescribed initial and boundary conditions (2.7), a single state variable is more applicable. For this, we assume θ and ϕ are single valued continuous functions of one another and rearrange the relation

$$(3.1) \quad \frac{\partial \theta}{\partial t} = \frac{\partial \theta}{\partial \phi} \frac{\partial \phi}{\partial t} = (\frac{1}{\frac{\partial \phi}{\partial \theta}}) \frac{\partial \phi}{\partial t}, \quad \frac{\partial \phi}{\partial \theta} = \frac{\partial \phi}{\partial h} \frac{\partial h}{\partial \theta}.$$

After differentiating (1.2) and (2.7) with respect to h, we have the following expressions

$$(3.2) \quad \frac{\partial \theta}{\partial h} = \alpha(\theta_s - \theta_r)(\alpha + |h|^\beta)^{-2}.\beta|h|^{\beta-1}, \quad \frac{\partial \phi}{\partial h} = \bar{K}(h) = K(\psi).$$

The transformed Richards Equation (2.6) with the help of (3.1) and (3.2) takes the following form

$$(3.3) \qquad c(\phi)\frac{\partial \phi}{\partial t} = \frac{\partial^2 \phi}{\partial z^2},$$

where the functional coefficient c depends on ϕ through h is described as

$$(3.4) \qquad c(\phi(h)) = \frac{\alpha\beta(\theta_s - \theta_r)|h|^{\beta-1}}{\bar{K}(h)(\alpha + |h|^{\beta})^2}.$$

3.1. **Finite Difference Approximation.** The standard finite difference approximation for the equation 3.3 starts with the supposition of $\Delta z = L/M$ and $\Delta t = T/N$. To apply this, firstly we construct a grid (z_j, t_n), with $z_j = j\Delta z$, $j = 0, 1, 2, \cdots, M$ and $t_n = n\Delta t$, $n = 0, 1, 2, \cdots N$. Let ϕ_j^n denote $\phi(z_j, t_n)$. Now (3.3) can be approximated using forward difference in time and central difference in space as

$$(3.5) \qquad \left.\frac{\partial \phi}{\partial t}\right|_{(z_j, t_n)} \approx \frac{\phi_j^{n+1} - \phi_j^n}{\Delta t}, \quad \left.\frac{\partial^2 \phi}{\partial z^2}\right|_{(z_j, t_n)} \approx \frac{\phi_{j-1}^n - 2\phi_j^n + \phi_{j+1}^n}{\Delta z^2}.$$

Let $0 \leq \delta \leq 1$. Using a weighted average of the derivative $\frac{\partial^2 \phi}{\partial z^2}$ at two time levels, t_n and t_{n+1}, the equation (3.3) can be discretized as:

$$(3.6) \qquad \phi_j^{n+1} = \phi_j^n + \lambda_j^n \left[\delta(\phi_{j-1}^{n+1} - 2\phi_j^{n+1} + \phi_{j+1}^{n+1}) + (1 - \delta)(\phi_{j-1}^n - 2\phi_j^n + \phi_{j+1}^n)\right],$$

where $\lambda_j^n = \Delta t/(c_j^n \Delta z^2)$.

Equation (3.6) is used to update the values of ϕ_j^{n+1} for the internal nodes. Using the constant pressure head (Dirichlet boundary) at the upper and lower boundaries, we get

$$(3.7) \qquad \begin{cases} \phi_0^{n+1} = \bar{\beta}_1(t_{n+1}), \\ \phi_M^{n+1} = \bar{\beta}_2(t_{n+1}). \end{cases}$$

The numerical scheme (3.6)-(3.7) represents a forward in time central in space (Forward Euler), backward in time central in space (Backward Euler) and Crank-Nicolson (CN) schemes for $\delta = 0$, $\delta = 1$ and $\delta = \frac{1}{2}$ respectively [15]. The error associated with this approximation is $\mathcal{O}(\Delta z^2 + \Delta t)$ for all $\delta \neq \frac{1}{2}$. In the case of Crank-Nicolson, it is $\mathcal{O}(\Delta z^2 + \Delta t^2)$, second order accurate in both space and time.

The corresponding tridiagonal matrix system of the above numerical procedure is

$$(3.8) \quad \begin{bmatrix} 1 & 0 & 0 & \ldots & 0 & 0 & 0 \\ -\delta\lambda_1^n & 1+2\delta\lambda_1^n & -\delta\lambda_1^n & \ldots & 0 & 0 & 0 \\ \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\ \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\ 0 & 0 & 0 & \ldots & -\delta\lambda_{M-1}^n & 1+2\delta\lambda_{M-1}^n & -\delta\lambda_{M-1}^n \\ 0 & 0 & 0 & \ldots & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \phi_0^{n+1} \\ \phi_1^{n+1} \\ \ldots \\ \ldots \\ \phi_{M-1}^{n+1} \\ \phi_M^{n+1} \end{bmatrix}$$

$$= \begin{bmatrix} \bar{\beta}_1(t_{n+1}) \\ (1-\delta)\lambda_1^n\phi_0^n + (1-2(1-\delta)\lambda_1^n)\phi_1^n + (1-\sigma)\delta_1^n\phi_2^n \\ \ldots \\ \ldots \\ \ldots \\ (1-\delta)\lambda_{M-1}^n\phi_{M-2}^n + (1-2(1-\delta)\lambda_{M-1}^n)\phi_{M-1}^n + (1-\delta)\lambda_{M-1}^n\phi_M^n \\ \bar{\beta}_2(t_{n+1}) \end{bmatrix}.$$

The numerical scheme (3.8) can be used to update the transformed variable ϕ_j^n to its value in the next time level ϕ_j^{n+1}. But we cannot advance the algorithm to the next time level ϕ_j^{n+2} without evaluating the function $c(\phi_j^{n+1})$ which requires computing the intermediate variable h_j^{n+1}. For this, we imply the equation (3.2) which can be approximated as

$$(3.9) \qquad\qquad h_j^{n+1} = h_j^n + \frac{\phi_j^{n+1} - \phi_j^n}{\bar{K}(h_j^n)}.$$

3.2. **Stability Analysis.** We use Von Neumann stability analysis for equation (3.6). We assume a Fourier component as $\phi_j^n = \bar{\phi}^n e^{ij2\pi h\zeta}$. Now the scheme (3.6) becomes

$$(3.10) \quad -\delta\lambda_j^n(e^{-2i\pi h\xi} - 2 + e^{2i\pi h\xi})\phi^{\bar{n}+1} + \phi^{\bar{n}+1} = (1-\delta)\lambda_j^n(e^{-2i\pi h\xi} - 2 + e^{2i\pi h\xi})\bar{\phi}^n + \bar{\phi}^n.$$

Using the relation

$$(3.11) \qquad\qquad 2h\pi\xi = u, \; cosu = \frac{e^{iu} + e^{-iu}}{2},$$

equation (3.10) becomes

$$(3.12) \quad \bar{\phi}^{n+1}(1 - \delta\lambda_j^n(e^{-2i\pi h\xi} - 2 + e^{2i\pi h\xi})) = \bar{\phi}^n(1 + (1-\delta)\lambda_j^n(e^{-2i\pi h\xi} - 2 + e^{2i\pi h\xi})).$$

Introducing an amplification factor such that $\bar{\phi}_j^{n+1} = G\bar{\phi}_j^n$, then

$$(3.13) \qquad\qquad \bar{\phi}^{n+1}(1 + 2\delta\lambda_j^n(1 - \cos u)) = \bar{\phi}^n(1 - 2(1-\delta)\lambda_j^n(1 - \cos u)),$$

gives

$$(3.14) \qquad\qquad G = \frac{(1 - 2(1-\delta)\lambda_j^n(1 - \cos u))}{(1 + 2\delta\lambda_j^n(1 - \cos u))}.$$

For a stable solution the absolute value of G must be bounded for all values of u. Mathematically, it is expressed as

$$|G| \leq 1,$$

$$\left|\frac{(1 - 2(1 - \delta)\lambda_j^n(1 - \cos u))}{(1 + 2\delta\lambda_j^n(1 - \cos u))}\right| \leq 1,$$

so that

$$(3.15) \qquad -1 \leq \frac{(1 - 2(1 - \delta)\lambda_j^n(1 - \cos u))}{(1 + 2\delta\lambda_j^n(1 - \cos u))}.$$

Inequality (3.15) is satisfied for all values of u. With the maximum value of $(1 - \cos u) = 2$ equation (3.15) takes the form

$$\frac{(1 - 4(1 - \delta)\lambda_j^n}{1 + 4\delta\lambda_j^n} \geq -1,$$

which is same as

$$(3.16) \qquad 4\lambda_j^n(1 - 2\delta) \leq 2,$$

it is easy to see that if $\delta \geq \frac{1}{2}$, inequality (3.16) is always satisfied and if $\delta < \frac{1}{2}$, inequality (3.16) is satisfied only if

$$(3.17) \qquad \lambda_j^n \leq \frac{1}{(1 - 2\delta)}$$

It can be shown that the numerical scheme (3.6) is unconditionally stable for $\delta \geq \frac{1}{2}$, and for $\delta < \frac{1}{2}$ the scheme is stable only if $\max \lambda_j^n \leq \frac{1}{2}$ which puts a severe restriction on the time step-size known as Courant–Friedrichs–Lewy (CFL) condition as

$$(3.18) \qquad \Delta t \leq \frac{1}{2}\left[\min c(\phi_j^n)\Delta z^2\right].$$

3.3. Super Time-Stepping Scheme.

When $\delta = 0$, the equation (3.8) takes the form

$$\begin{bmatrix} \phi_0^{n+1} \\ \phi_1^{n+1} \\ ... \\ ... \\ \phi_{M-1}^{n+1} \\ \phi_M^{n+1} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & ... & 0 & 0 & 0 \\ \lambda_1^n & 1 - 2\lambda_1^n & \lambda_1^n & ... & 0 & 0 & 0 \\ ... & ... & ... & ... & .... & .... & .... \\ ... & ... & ... & ... & .... & .... & .... \\ 0 & 0 & 0 & ... & \lambda_{M-1}^n & 1 - 2\lambda_{M-1}^n & \lambda_{M-1}^n \\ 0 & 0 & 0 & ... & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \phi_0^n \\ \phi_1^n \\ ... \\ ... \\ \phi_{M-1}^n \\ \phi_M^n \end{bmatrix} + \begin{bmatrix} \bar{\beta}_1(t_{n+1}) \\ 0 \\ ... \\ ... \\ 0 \\ \bar{\beta}_2(t_{n+1}) \end{bmatrix},$$

and it gives the following expression

$$(3.19) \qquad \Phi^{n+1} = R(\Delta t A)\Phi^n + \Delta t E^n,$$

where the amplification factor $R = I - \Delta t A$ with the tridiagonal matrix A appropriately defined from the above linear system.

It is clear that the right hand side of this equation (3.19) is explicitly known at the time level t_n and there is no need to matrix inversion or not to apply any iterative approach to solve it. But, there is a major drawback of this scheme. A stumbling block is appear by the stability criterion, also known as CFL–condition (3.9).

As is shown the above algorithm (3.18) is subjected to the restrictive stability condition which is equivalent to the fact that the spectral radius of matrix $R \leq 1$.

Super Time–Stepping scheme alleviates the restriction of the CFL condition by requiring stability at the end of one Super Time-Step $\Delta \tau$ consisting of a cycle of s sub–steps, rather than at the end of each time step Δt, thus leading to a Runge–Kutta like method with s stages. The interior values have, in general, no approximation properties and it should be considered as intermediate calculations. The Runge–Kutta–Legendre(RKL) methods are focused on using shifted Legendre polynomials as the stability polynomial of the scheme. Like wise Chebyshev polynomials, Legendre polynomials are bounded in unity and they are useful to develop a stable scheme. Here we use a stabilized Runge–Kutta–Legendre Super Time-Stepping scheme where the amplification factor $R_s(\Delta \tau A)$ is defined in terms of the Legendre polynomials [14]

$$(3.20) \qquad R_s(z) = a_s + b_s P_s(w_o + w_1 z).$$

The parameters $a_s = 0$, $b_s = 1$, $w_0 = 1$ and $w_1 = \frac{2}{s^2+s}$ are chosen to satisfy the consistency conditions at first order $R_s(0) = 1$ and $R'_s(0) = 1$. Thus the s-stage RKL scheme takes the form

$$(3.21) \qquad \Phi^{n+1} = P_s(I - w_1 \Delta \tau A)\Phi^n + \Delta \tau E^n,$$

from the three point recursion property of the Legendre polynomials

$$(3.22) \qquad (k)P_k(x) = (2k-1)xP_{k-1}(x) - (k-1)P_{k-2}(x).$$

The RKL scheme (3.20) can be written as

$$(3.23) \qquad \begin{cases} U_0 = \Phi^n \\ U_1 = U_0 - \bar{\mu}_1 \Delta \tau A U_0 + \Delta \tau E^n \\ U_k = \mu_k U_{k-1} + \nu_k U_{k-2} - \bar{\mu}_k \Delta \tau A U_{k-1}, \quad 2 \leq k \leq s \\ \Phi^{n+1} = U_s \end{cases},$$

where

$$\mu_k = \frac{2k-1}{k}, \quad \bar{\mu}_k = \frac{2k-1}{k}w_1, \quad \nu_k = \frac{1-k}{k}.$$

We have that the RKL scheme has a maximum super-step of

$$\Delta \tau_{max} = \frac{\Delta t}{w_1} = \Delta t \left(\frac{s^2+s}{2} \right).$$

The Runge–Kutta–Legendre method described from above recursive relation is consistent and stable at each of the intermediate stages which is more applicable for output.

4. Simulation Results

4.1. **Numerical Setup.** The numerical manual developed in the previous section is written in Python, and ran on a pc with 2.50 GHz Quad–Core Intel Core i5 processor. We analyzed the behavior of the four numerical schemes presented in the previous section in a specific infiltration experiment. In this simulation we considered a vertical soil column of depth $L = 70$ cm in a time period of $t_{max} = 1$ hr.

Following Haverkamp et al. [4], we used the soil parameters and the characteristics relationship between the soil moisture content $\theta(\psi)$ and the hydraulic conductivity $K(\psi)$ as follows

$$\theta(\psi) = \theta_r + \frac{\alpha(\theta_s - \theta_r)}{\alpha + |\psi|^\beta},$$

$$K(\psi) = K_s \frac{A}{A + |\psi|^\gamma}.$$

The simulation starts with a uniform saturation and a constant water head $\psi = -61.5$ cm is maintained at the bottom and top boundary $z = 0, z = L$.

To compute the approximate solution using the explicit scheme i.e. Forward Euler, we have used a uniform spatial step–size $\Delta z = 1$ cm and the time step–size $\Delta t = 0.005$ sec which guarantees the CFL condition (3.18). This being a highly resolved in temporal direction, we use it as the representative solution to compare with all other methods. The implicit schemes i.e. Backward Euler and Crank–Nicolson (CN), being unconditionally stable, have no restriction on the time step–size. The step–size is estimated only by the accuracy of approximation. We developed the stabilized Runge–Kutta–Legendre Super Time–Stepping (RKL) scheme which is also unconditionally stable. The step–size in RKL is defined by the appropriateness of number of sub–steps in one Super Time–Stepping cycle and, of course, this can be chosen by compromising the accuracy of approximation. As such, to analyze the performance of RKL compared to the implicit schemes Backward Euler and Crank–Nicolson, we run Backward Euler and Crank–Nicholson with $\Delta t = \sum_{k=1}^{s} \Delta\tau$ for different values of s. To deal with the nonlinear dependence of the functional coefficient $c(\phi)$ in the implicit schemes, we use fix point iteration with maximum allowable iteration $MAXIT = 10$ and relative error tolerance $TOL = 10^{-6}$.

4.2. **Results and Discussion.** In general, Richards Equation is a highly nonlinear degenerate PDE. The nonlinear behavior appears from the use of constitutive relationship between θ and ψ and K and ψ. The numerical solutions of the Richards Equation is computed using four different schemes namely Forward Euler, Backward Euler, Crank–Nicolson and RKL. First, we observed that Forward Euler scheme is conditionally stable where as Backward Euler, Crank–Nicolson and RKL schemes are unconditionally stable. Thus, to create a long simulation with the explicit scheme, it is necessary to fix the CFL–satisfying time step–size. Since we do not have accurate solution to estimate the errors, we use the numerical solution obtained from the fully explicit numerical scheme with a fine mesh ($\Delta z = 1$ cm, and $\Delta t = 0.002$ sec) as the reference solution.

For the comparison of the methods regarding to their performance, we make simulations with a fairly fine mesh of $\Delta z = 2$ cm, and find Δt_{expl} using CFL criteria and define $\Delta\tau = \Delta t_{expl}\left(\frac{s^2+s}{2}\right)$ as the time step size of a sub step in a super time step. Keeping in mind that, comparison of explicit and implicit scheme is not reliable and authentic. Explicit scheme require too many time-steps and implicit scheme requires a large number of iterations. As we are using linear system solver in the implicit schemes, it is impossible to compare the efficiency using the same time-step size. To make the comparison more reliable, we have used the time step–size for implicit schemes equal the duration of one
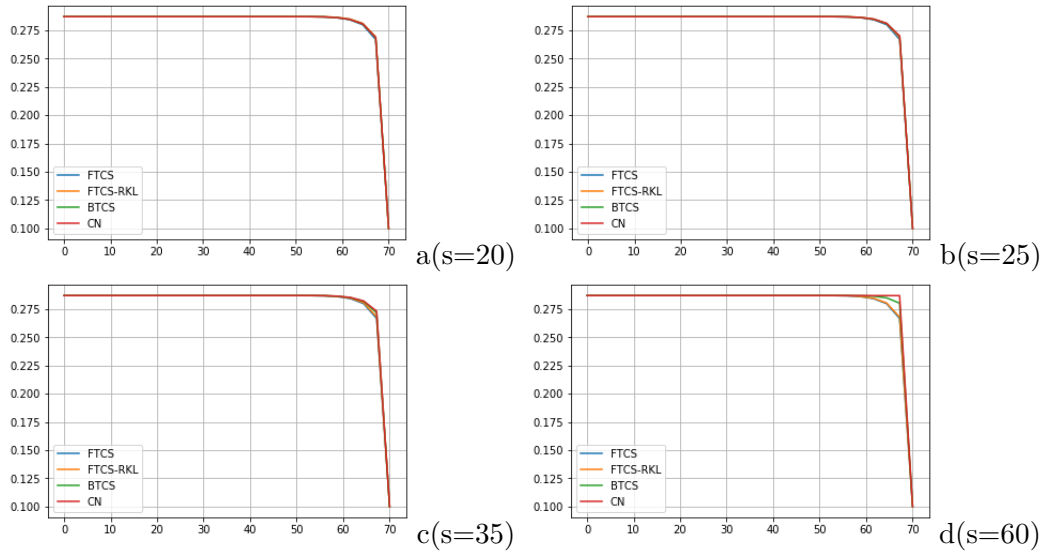
FIGURE 1. Convergence of methods [Forward Euler, RKL, Backward Euler, Crank-Nicolson] s = 20 in (a), s = 25 in (b), s = 35 in (c) and s = 60 in (d)
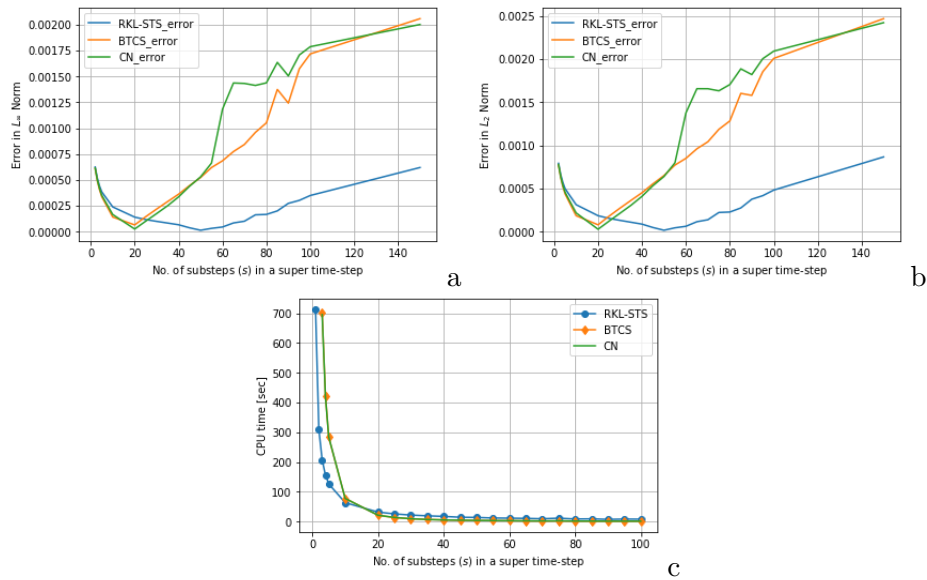
.

FIGURE 2. Test comparison of the four methods error in infinity norm(a), in 2 norm (b) and speedup with RKL(c)

.

super–step in RKL, $\Delta t = s\Delta\tau$, the time period of one super time step size. The simulation results (speedup and the accuracy) are mention in the Tables 1–3 and Figures 1–3. The corresponding CPU time (in sec) of all the numerical schemes implemented in this work is shown in the Table 1. The total CPU time of the RKL scheme for different values of s shows the acceleration in the speed of the explicit scheme Forward Euler ($s = 1$). Figure 1, shows the comparison of the CPU timings of the RKL for various values of s with the implicit schemes Backward Euler and Crank-Nicolson. The errors comparison in the final values of water moisture content θ at t=1 hr obtained from RKL and the other three schemes in
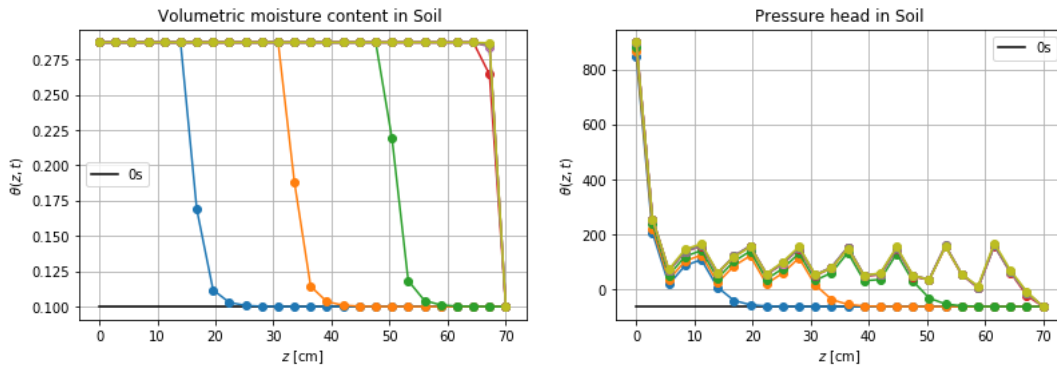
Figure 3. Variational trend of moisture content in depth (left) and pressure head (right)

.

Table 1. Timing of four Methods

s	RKL_Time	Impl_$Time$	CN_$Time$
1	635.336	2496.323	2477.082
2	287.616	867.910	864.529
3	188.661	441.445	440.695
4	142.378	268.695	269.015
5	113.126	180.680	184.741
10	56.743	50.773	51.763
20	28.572	13.651	13.561
25	23.451	8.991	8.941
30	19.991	6.450	6.310
35	17.261	5.270	6.480
40	14.901	3.790	4.660
45	13.041	2.970	4.330
50	12.891	2.200	3.570
55	11.021	2.100	1.800
60	10.351	1.810	2.990
65	9.891	1.600	2.400
70	8.740	1.440	2.090
75	9.281	1.460	2.010
80	9.821	1.590	1.720
85	8.510	1.270	1.680
90	6.920	1.080	1.440
95	7.220	0.940	1.420
100	8.000	1.030	1.460
150	7.360	0.740	1.050

TABLE 2. Test of RKL scheme with Forward Euler, Backward Euler and Crank–Nicolson in L^∞

s	$||RKL - EE||_\infty$	$||RKL - IE||_\infty$	$||RKL - CN||_\infty$
1	7.91883E-07	7.92191E-07	7.92037E-07
2	4.16374E-04	4.15648E-04	4.15646E-04
3	3.44738E-04	3.42923E-04	3.42917E-04
4	2.94148E-04	2.90884E-04	2.90868E-04
5	2.56587E-04	2.51515E-04	2.51480E-04
10	1.57518E-04	1.38127E-04	1.37661E-04
20	9.15400E-05	2.00068E-05	1.82310E-05
25	7.70383E-05	3.25982E-05	4.89687E-05
30	6.73398E-05	8.49890E-05	1.18264E-04
35	6.05417E-05	1.38871E-04	1.98461E-04
40	5.55521E-05	1.94192E-04	2.95403E-04
45	5.18694E-05	2.50644E-04	6.76712E-04
50	4.86540E-05	4.91707E-04	7.59637E-04
55	4.59088E-05	5.78174E-04	7.63081E-04
60	4.41741E-05	4.98196E-04	7.64781E-04
65	4.35869E-05	7.65184E-04	7.17074E-04
70	3.88258E-05	7.70164E-04	7.70164E-04
75	3.91283E-05	6.00136E-04	7.69862E-04
80	4.21220E-05	7.64327E-04	7.66867E-04
85	4.45288E-05	7.64461E-04	7.61731E-04
90	2.77711E-05	7.81219E-04	7.54348E-04
95	5.20295E-05	7.56960E-04	7.56504E-04
100	1.86392E-05	7.90351E-04	7.90351E-04
150	1.93465E-04	1.00245E-03	9.86502E-04

L_∞-norm and L_2-norm are in Tables 2 and 3. Also table 2 shows the stability of the explicit scheme RKL as the overall error in L_∞-norm with respect to the reference solution is not increasing much with the increase in the values of s. From Table 3, similar behavior are observed with the errors measured in L_2-norm.

Comparison of the solution $\theta_s^{RKL}(z,t)$ at $t = 0.7hr$ with different values of s to fully explicit $\theta^{ForwardEuler}(z,.7)$, implicit $\theta^{BackwardEuler}(z,.7)$ and Crank-Nicolson $\theta^{CN}(z,.7)$ methods are shown in Figure 3. These values of s are selected randomly to show that as s becomes large, there will be less accurate approximation and as $s \to 1$ the method becomes the most accurate. From the Figure 3, we can see that $s = 25$ gives the best performance. It should be pointed out that RKL can be run with higher s having low accuracy. Finally, we present some numerical simulation results in a very fine spatial mesh $\Delta z = 0.5$ cm obtained from the RKL method ($s = 60$) and CN ($\Delta t = s\Delta\tau$) . In Figures 3 and 4, we describe the profile

TABLE 3.   Test of RKL scheme with Forward Euler, Backward Euler and Crank–Nicolson in L^2

s	$||RKL-EE||_2$	$||RKL-IE||_2$	$||RKL-CN||_2$
1	8.98269E-07	8.98580E-07	8.98425E-07
2	4.77095E-04	4.76305E-04	4.76304E-04
3	3.98887E-04	3.96918E-04	3.96912E-04
4	3.42530E-04	3.38994E-04	3.38978E-04
5	3.00130E-04	2.94642E-04	2.94607E-04
10	1.86165E-04	1.65270E-04	1.64810E-04
20	1.08531E-04	3.29225E-05	2.78803E-05
25	9.12225E-05	3.32430E-05	4.96658E-05
30	7.95618E-05	8.89504E-05	1.23535E-04
35	7.13300E-05	1.48150E-04	2.10045E-04
40	6.52825E-05	2.09086E-04	3.14908E-04
45	6.07926E-05	2.71179E-04	7.23534E-04
50	5.68181E-05	5.29720E-04	8.08834E-04
55	5.38729E-05	6.19700E-04	8.12438E-04
60	5.14025E-05	5.43346E-04	8.14686E-04
65	5.01292E-05	8.02862E-04	7.69098E-04
70	4.51372E-05	8.03892E-04	8.20662E-04
75	4.37948E-05	6.65214E-04	8.19488E-04
80	4.94853E-05	8.12688E-04	8.16599E-04
85	4.94136E-05	8.01545E-04	8.13324E-04
90	3.14526E-05	8.34195E-04	8.09112E-04
95	5.60425E-05	8.09116E-04	8.08688E-04
100	2.01036E-05	8.44869E-04	8.44699E-04
150	2.51358E-04	1.10339E-03	1.08891E-03

of moisture content $\theta(z,t)$ and pressure head $\psi(z,t)$ along the soil depth at various times $t = 0.1, 0.2, \cdots, 1$ hr.

5. CONCLUSION

In this paper, we considered a highly nonlinear degenerate parabolic partial differential equation (Richards Equation in 1D) and solved it numerically. We apply different finite difference schemes like Forward Explicit, Backward Implicit, Crank–Nicolson, and a Stabilized Runge–Kutta–Legendre Super Time–Stepping scheme RKL. The numerical simulations show that the RKL scheme boasts large efficiency gains compared to the standard Forward Explicit scheme and is comparable to Crank–Nicolson. Forward schemes are simple in nature and give accurate implementation and are parallelizable but suffer severely from stability restriction on the time step–size. In another side, Implicit schemes involve iterative

approach to solve the nonlinear systems, hence are moderately efficient, but computationally very costly for highly nonlinear problems and comparatively difficult to implement. From computational point of view, Super Time–Stepping is more efficient than the standard Forward implicit schemes, in that it runs at least as fast with better accuracy and it is much easier to program also it can be easily extended to problems in higher dimensions.

Data Availability

The data used for supporting the findings of this study are included within the article.

Conflicts of Interest

The authors declare that there are no conflicts of interest.

References

[1] Richards, L. A., "Capillary conduction of liquids through porous mediums",*Physics 1*, (5), 318-333. 1931.

[2] Van Genuchten, M.Th., E.A. Sudicky, "Recent Advances in Vadose Zone flow and transport modeling, Vadose Zone Hydrology cutting across disciplines", *Oxford University Press*, 155-193, 1999.

[3] Celia, M. A. and E. T. Bouloutas, "A General Mass–Conservative Numerical solution for the Unsaturated flow equation", *Water Resource research*, 26, 1483-1490, 1990.

[4] Haverkamp, R., M. Vauclin, J. Touma, P. J. Wierenga, G. Vachaud, "A comparison of numerical simulation models for one-dimensional infiltration", *Soil Sci. Soc. Am. J.*, 41(2),285–294, 1997.

[5] Berganaschi, L., PUTTI, M., "Mixed finite elements and Newton-type militarizations for the solution of the Richards equations", *Int.J. Numer. Methods Eng.*, 45, 1025-1046, 1999.

[6] Eymard, R., Gutnic, M., Hillhorst, D., "The finite volume method for richard equation", textitCompute. Geosci, 3, 259-294, 1999.

[7] Gottardi, G. and M. Venuteli, "Richards: Computer Programming for the numerical simulation of one-dimensional infiltration into unsaturated soil",*Computers and Geo-sciences*, 19, 1239-1266, 1993.

[8] Liu, F., Fukumoto, Y., Zhao, X., "A linearized finite difference Scheme for the Richards equation under variable-Flux boundary conditions",*Journal of scientific computing*, 83, 2-21, 2020.

[9] Zambra,C.E., Dumbser, M., Toro , E.F. and Morago, N.O., "A novel numerical method of high-order accuracy for flow in unsaturated porous media". *Int. J. Numer. Meth. Engng*, 89, 227-240,2012.

[10] Kavetsky, D., Binning, P., Sloan, S.W., " Adaptive time stepping and error control in a mass conservative numerical solution of the mixed form of Richards equation", *Adv. Water Res.*, 24, 595-605, 2001.

[11] Radu, F. A. and Florian List, "A Study on iterative methods for solving Richards equation", *Computers and Geo-sciences*, 20,341-353, 2016.

[12] Egid,N., E. Gioia, P. Maponi and L. Spadoni., "A numerical solution of Richards equation: a simple method adaptable in parallel computing", *International Journal of computer Mathematics*, 97, 2-17, 2020.

[13] Timsina R.C, Khanal H., Uprety K.N,(2021) "An explicit stabilized Runge-Kutta-Legendre super time -stepping scheme for the Richards equation", *Mathematical problems in Engineering*, vol-2021, 1-11, 2021

[14] Chad D. Meyer, Dinshaw S. Balsara, Tariq D. Aslam, "A stabilized Runge-Kutta-Legendre method for explicit super-time-stepping of paraboliic and mixed equations", *Journal of Computational Physics*, 257, 594-626, 2014.

[15] J.W. Thomas, "Numerical partial differential equations: Finite difference methods", Text in Applied Mathematics, Springer, 2010.

[16] V. Alexiades, V. Amiez, P.-A., Gremaud, "Super-Time-Stepping acceleration of explicit schemes for parabolic problems", Communications in Numerical Methods in Engineering, 12, 31-42, 1996.