Adaptation and Validation of the ParSWMS Numerical Code for Simulation of Water Flow and Solute Transport in Soilless Greenhouse Substrates

Mohammad R. Gohardoust[1], Jirka Šimůnek[2], Horst Hardelauf[3], and Markus Tuller[1*]

[1]Department of Environmental Science, The University of Arizona, Tucson, AZ, USA
[2]Department of Environmental Sciences, University of California, Riverside, CA, USA
[3]Institute of Bio-and Geosciences, IBG-3 Agrosphere, T, Jülich, Germany

Journal of Hydrology

***Corresponding Author**

Markus Tuller, Ph.D.
Professor of Soil & Environmental Physics
Department of Environmental Science
The University of Arizona
Phone: (520) 621-7225
Email: mtuller@arizona.edu

Highlights

- The ParSWMS code was adapted to simulate flow and transport in soilless substrates
- The functionality of ParSWMS was verified using HYDRUS (2D/3D) as reference
- High performance computing was utilized to run the ParSWMS
- Preconditioner/solver combinations were tested to increase computational efficiency

Keywords

- Soilless culture
- Water flow and solute transport
- Numerical simulations
- High performance computing
- HYDRUS (2D/3D)
- ParSWMS

Abstract

Numerical simulation of three-dimensional water flow and solute transport in containerized variably saturated soilless substrates with complex hydraulic properties and boundary conditions necessitates high-resolution discretization of the spatial and temporal domains, which commonly leads to several million nodes requiring numerical evaluation. Even today's computing prowess of workstations is not adequate to tackle such problems within a reasonable timeframe, especially when numerous realizations are required to optimize the geometry, substrate properties, and irrigation and fertigation management of soilless plant growth modules. Hence, the parallelization of the numerical code and utilization of high performance computing (HPC) are essential. Here, we adapted and applied the ParSWMS parallelized code that is amenable to solving the 3D Richards equation for water flow and the convection-dispersion equation for solute transport subject to linear solute adsorption. The code was modified to allow for nonlinear equilibrium solute adsorption with new boundary conditions and applied to simulate water flow and nitrogen and phosphorus transport in containerized soilless substrates. Multi-solute transport simulations with the modified Linux ParSWMS code were first performed on a workstation and referenced to the Windows-based HYDRUS (2D/3D) numerical code. After confirming the agreement between the modified ParSWMS code and HYDRUS (2D/3D), various preconditioners and iterative solvers were evaluated to find the computationally most efficient combinations. The performance of the modified ParSWMS code and its stability were compared to HYDRUS (2D/3D) simulations for three soilless substrates consisting of horticultural perlite, volcanic tuff, and a volcanic tuff/ coconut coir mixture. Considering the solute mass balance error as a stability measure, ParSWMS outperformed HYDRUS (2D/3D). Moreover, simulations with the modified ParSWMS code were about 22% faster than simulations with HYDRUS (2D/3D) on the workstation. Tests of the modified ParSWMS on two HPC clusters with 28 and 94 cores revealed a potential computational speedup of 94% relative to the HYDRUS (2D/3D) simulations performed on the workstation.

1. Introduction

Because of the urgent need to secure and sustain the food and water supply for an ever-growing human population, especially in underdeveloped arid and semiarid regions of the world, and an increasing demand for out-of-season fruits, vegetables, and ornamentals in the industrial world, there is a momentous incentive to shift from soil-based crop production to more resource-efficient containerized soilless production systems (*Raviv et al.*, 2019). Soilless substrates exhibit major advantages over soils. Besides the alleviated risk for spreading soilborne pathogens, their physical and hydraulic properties (i.e., plant water availability after irrigation and the aeration characteristics) are superior to those of arable soils (*Savvas*, 2003; *Savvas and Gruda*, 2018). Furthermore, the nutrient availability to plant roots can be better controlled in soilless substrates, which commonly leads to healthier plants and higher yields than in soil-based production (*Raviv et al.*, 2019). It is even possible to tailor (i.e., engineer) substrates that exhibit specific physical and hydraulic properties beneficial for the growth of different crops. Though the same physical principles apply to both soilless substrates and soils, their physical and hydraulic properties are vastly different.

In addition, there are fundamental differences with regard to dynamic water, air, and nutrient distribution processes and root growth and development between spatially confined growth containers and unconfined field soils. While an impermeable container bottom (with drainage holes) restricts water flow and nutrient transport in growth containers, water drains and redistributes to much deeper layers in agricultural soils unless natural impediments exist. This leads to vastly different infiltration and redistribution dynamics, requiring more intensive management of soilless systems (*Gohardoust et al.*, 2020).

Taking into consideration that currently most of the soilless culture plant growth experiments are based on trial and error, three-dimensional (3D) numerical flow and transport simulations can provide a scientifically sound basis for a priori elimination of substrates with unfavorable growth potential, as well as guidance for growth module design and irrigation and fertigation management, thereby preventing costly mistrials.

Today, numerical computer codes that solve the governing 3D water flow and solute transport equations are indispensable for advancing the understanding of complex porous media processes. However, intricate hydraulic properties and boundary conditions demand high-resolution discretization of the spatial and temporal domains, often leading to several million nodes requiring numerical evaluation (*Šimůnek et al.*, 2016). Even the current computing prowess of workstations is not adequate to tackle such simulations within a reasonable timeframe, especially when numerous realizations are desired to optimize the geometry, soilless substrate properties, and irrigation and fertigation management of plant growth modules. Hence, parallelization of the numerical codes and utilization of high-performance computing (HPC) clusters are vital.

Hardelauf et al. (2007) were among the first to develop a parallelized code for 3D simulation of water flow and solute transport that they termed ParSWMS. The code is based on the SWMS-3D model of *Šimůnek et al.* (1995), which numerically solves the Richards equation for saturated/unsaturated water flow and the convection-dispersion equation (CDE) for solute transport using a Galerkin-type linear finite element scheme. The resulting nonlinear partial differential equation (PDE) for water flow and the linear PDE for the CDE (i.e., assuming linear adsorption isotherms) are solved in SWMS-3D with preconditioned conjugate gradient and Orthomin (*Vinsome*, 1976) methods, respectively. In ParSWMS, *Hardelauf et al.* (2007) employed ParMETIS, an open source MPI-based library, to distribute the simulation to multiple processing

nodes, and the *Portable, Extensive Toolkit for Scientific Computation* (PETSc) library (https://www.mcs.anl.gov/petsc/) for preconditioning and solving the resulting system of linear equations.

Driven by the need to execute a vast number of 3D flow and transport simulations to aid with the design and management of containerized plant growth modules, we adopted ParSWMS and modified the code to: (1) permit the simultaneous application of time-variable flux and free drainage boundary conditions (BCs) and allow for the time-variable flux BC to be treated as an atmospheric BC; and (2) enable multi-solute transport simulations with nonlinear equilibrium adsorption. These modifications are essential for the simulation of surface drip irrigation and to account for the nonlinear adsorption behavior of phosphorus and ammonium.

Using HYDRUS (2D/3D) version 3.02.0560 (*Šimůnek et al.,* 2018) as a reference, we first meticulously verified the modified ParSWMS code for a simple, yet realistic, soilless growth module setup. Thereafter, we focused on computational efficiency and evaluated a vast number of preconditioner/solver combinations for both water flow and solute transport based on a real growth module used in production scale greenhouse experiments. Subsequently, we benchmarked the soilless substrate flow and transport simulation results against HYDRUS (2D/3D) on a workstation with both Windows and Linux OS installed. As a final step, we executed the modified ParSWMS code on two HPC clusters to demonstrate the significant computational speedup relative to the HYDRUS (2D/3D) simulations performed on the workstation. Please note that for brevity we refer to HYDRUS instead of HYDRUS (2D/3D) in the following.

2. Modifications to ParSWMS

2.1 Boundary Conditions

In the ParSWMS water flow simulation each node is assigned an integer code ranging from -6 to 6. While positive values are allocated to nodes with prescribed pressure heads, zero and negative values represent nodes with prescribed volumetric water fluxes. In the original ParSWMS code (*Hardelauf et al.*, 2007) both time-variable flux and free drainage boundary nodes are assigned -3. Hence, these BCs cannot be used simultaneously. To resolve this issue, the free drainage boundary condition code was set to -6, similar as in HYDRUS, which was not allocated in the original ParSWMS. The discharge rate at a free drainage node n is determined as $Q(n) = -width(n) \times K(h)$, where $width(n)$ is the surface area associated with node n, and $K(h)$ is the hydraulic conductivity as a function of the pressure head h.

To account for stage-I (i.e., potential evaporation rate controlled by atmospheric demand) and stage-II (i.e., falling evaporation rate limited by the ability of the substrate to transmit water to the surface) evaporation, it is necessary to treat the variable flux BC imposed for simulating surface drip irrigation as an atmospheric BC. The atmospheric BC that is implemented in both HYDRUS and ParSWMS is expressed as (*Neuman et al.*, 1974):

$$\begin{cases} \left| K\left(K_{ij}^{A}\frac{\partial h}{\partial x_j} + K_{iz}^{A}\right)n_i \right| \le E \\ \quad and \\ hCritA \le h \le hCritS \end{cases} \tag{1}$$

where K is the hydraulic conductivity, K_{ij}^{A} are the components of the anisotropy tensor $\boldsymbol{K}^{A}$, x_i (i=1,2,3) are the spatial coordinates, n_i are the components of the outward unit vector normal to the atmospheric boundary, E is the maximum potential evaporation rate, and $hCritS$ and $hCritA$

are two limiting values for the surface pressure head h. The Einstein summation convention is applied to Eq. (1), which implies summation over all possible values of the index in algebraic terms with repeated indices. While the $hCritS$ value specifies the maximum allowed pressure head at the soil surface (usually 0.0), the $hCritA$ value represents the minimum allowed surface pressure head defined based on equilibrium conditions between soil water and atmospheric vapor. The modified ParSWMS code now automatically switches between Dirichlet and Neumann boundary conditions for nodes with a variable flux BC if one of these limiting points is reached.

2.2 Multi-Solute Transport with Nonlinear Equilibrium Adsorption

To allow realistic simulations of phosphorus and ammonium transport processes in containerized soilless substrates, an option for nonlinear equilibrium solute adsorption was added to the modified ParSWMS code. The underlying CDE in ParSWMS is given as:

$$\frac{\partial \theta c}{\partial t} + \frac{\partial \rho s}{\partial t} = \frac{\partial}{\partial x_i}\left(\theta\, D_{ij} \frac{\partial c}{\partial x_j}\right) - \frac{\partial q_i c}{\partial x_i} + \mu_w \theta c + \mu_s \rho s + \gamma_w \theta + \gamma_s \rho - Sc \qquad (2)$$

where θ is the volumetric substrate water content [$L^3\,L^{-3}$], c is the solution concentration [$M\,L^{-3}$], s is the concentration adsorbed to the solid phase [$M\,M^{-1}$], ρ is the bulk density of the substrate [$M\,L^{-3}$], D_{ij} is the dispersion coefficient tensor [$L^2\,T^{-1}$], q_i is the i-th component of the volumetric flux [LT^{-1}], μ_w and μ_s are first-order rate constants for solutes in the liquid and solid phases [T^{-1}], respectively, γ_w and γ_s are zero-order rate constants for the liquid [$M\,L^{-3}\,T^{-1}$] and solid [T^{-1}] phases, respectively, S is the sink term in the water flow equation [T^{-1}], and c is the concentration of the sink term [$M\,L^{-3}$] with potential i and j values of 1, 2, 3. The equilibrium adsorption isotherm in its general form can be written as:

$$s = \frac{k_s c^{\beta}}{1 + \eta c^{\beta}} \tag{3}$$

where k_s [$L^{3\beta}M^{-\beta}$] is the distribution coefficient, and β [-] and η [$L^{3\beta}M^{-\beta}$] are coefficients of the nonlinear Freundlich and Langmuir adsorption isotherm models, respectively. For linear adsorption $\beta = 1$ and $\eta = 0$ with units of [$L^3 M^{-1}$]), for nonlinear Langmuir adsorption $\beta = 1$, and for nonlinear Freundlich adsorption $\eta = 0$.

Substituting Eq. (3) into Eq. (2) yields a nonlinear differential equation that can be solved with nonlinear iterative schemes. Picard iteration was chosen for the nonlinear CDE in the same fashion as for the Richards equation. Note that numerical solvers and preconditioners specific for nonsymmetric matrices are required for the discretized CDE (see Section 3 below). To increase the computational efficiency, simultaneous multi-solute transport simulations (i.e., phosphorus and ammonium) were enabled in the modified ParSWMS code – again, similar as in the current version of HYDRUS.

3. Hydraulic and Chemical Soilless Substrate Properties

The selection of the three soilless substrates that were considered for the simulations was guided by production scale greenhouse tomato growth experiments at the Volcani Center in Israel. The substrates included perlite, volcanic tuff, and a 70/30 vol.-% volcanic tuff/coconut coir mixture.

Horticultural perlite (Fig. 1a) is a natural amorphous volcanic glass that is formed through the hydration of obsidian. Perlite is commonly heated to 1000˚C, which causes structural water to evaporate and the volume to expand to about 4 to 20 times of its original size when rehydrated (*Bar-Tal et al.*, 2019). The sieved perlite aggregates are light weight, inert, and pathogen free *(Noland et al.*, 1992).

Tuff (Fig. 1b) is a pyroclastic volcanic material with high porosity and surface area. The mineral composition and weathering stage, in conjunction with physical alterations (i.e., grinding and sieving), determine its physicochemical properties (*Silber et al.*, 1994). Tuff possesses a high buffering capacity and may adsorb and release nutrients throughout the growing season, especially phosphorus (*Silber et al.*, 1999; *Silber and Raviv*, 1996). It exhibits bulk densities between 0.8 and 1.5 g cm^{-3} and a total porosity ranging from 60 to 80%.

Coconut coir consists of short and medium-length fibers of the mesocarp of Cocos nucifera L. that are left from various industrial applications. It exhibits remarkable physical and chemical characteristics such as high water holding capacity, good drainage and aeration properties, and high cation exchange capacity (*Evans et al.*, 1996; *Abad et al.*, 2005). It is commonly used as a surrogate for peat moss and mixed with mineral substrates such as tuff (Fig. 1c).

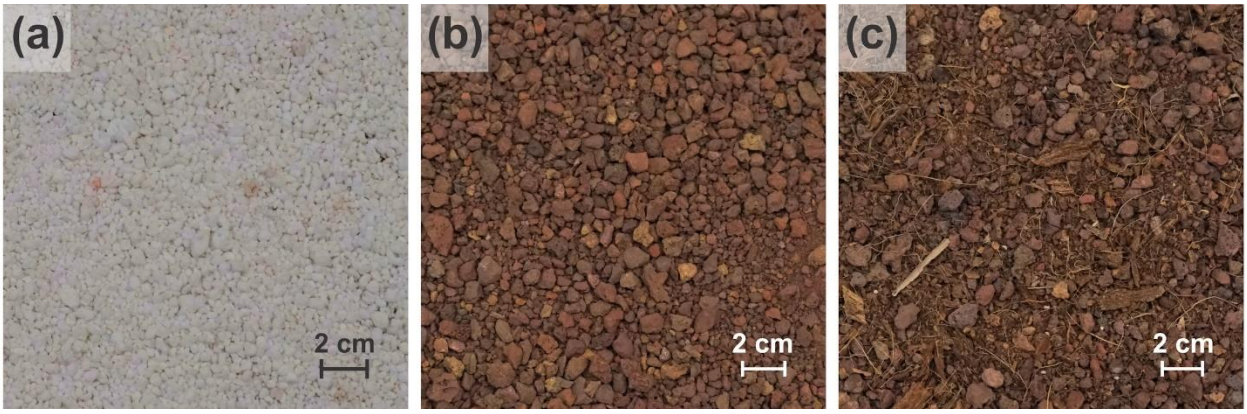

Fig. 1. Soilless substrates considered for flow and transport simulations. (a) Perlite, (b) tuff, and (c) 70/30 vol.-% volcanic tuff/coconut coir mixture.

The substrate water characteristics (SWC) were measured with Tempe cells (Soilmoisture Equipment Corp., Santa Barbara, CA) connected to a pressure manifold with a high-resolution pressure/vacuum regulator and gauge. Initially saturated samples were sequentially desaturated via application of increasing pressures after each equilibration phase. The saturated hydraulic

conductivities (K_{sat}) were measured with an automated constant head permeameter. Details about the SWC and K_{sat} measurements are provided in *Gohardoust et al.* (2020).

After thoroughly testing several SWC models comprising the *van Genuchten* (1980) model, a modified version of the *van Genuchten* model (*Vogel and Císlerová*, 1988; *Vogel et al.*, 2000), and the *Brooks and Corey* (1966) model, we found that the latter was best suited for mitigation of numerical instability issues caused by the extreme nonlinearity of the hydraulic conductivity function near saturation for all considered soilless substrates (Fig. 1). The *Brooks and Corey* (BC) SWC model is given as:

$$\begin{cases} \theta(h) = \theta_r + (\theta_s - \theta_r)\left(\frac{h_b}{h}\right)^{\lambda} & h < h_b \\ \theta(h) = \theta_s & h \geq h_b \end{cases} \quad (4)$$

$$K(h) = K_{sat}\left(\frac{h}{h_b}\right)^{-2-3\lambda} \quad (5)$$

where h is the pressure head, h_b is the air entry pressure (i.e., pressure threshold related to the onset of drainage of the largest pore in the system – a transition from fully to partially saturated conditions), θ is the water content expressed as a function of h, θ_s is the saturated water content, θ_r is the residual water content, λ is an empirical shape parameter, and K is the hydraulic conductivity expressed as a function of h.

The parameters of the BC model were determined via least-square fitting to the measured $\theta - h$ data pairs (Fig. 2), with SWC parameters and K_{sat} listed in Table 1.

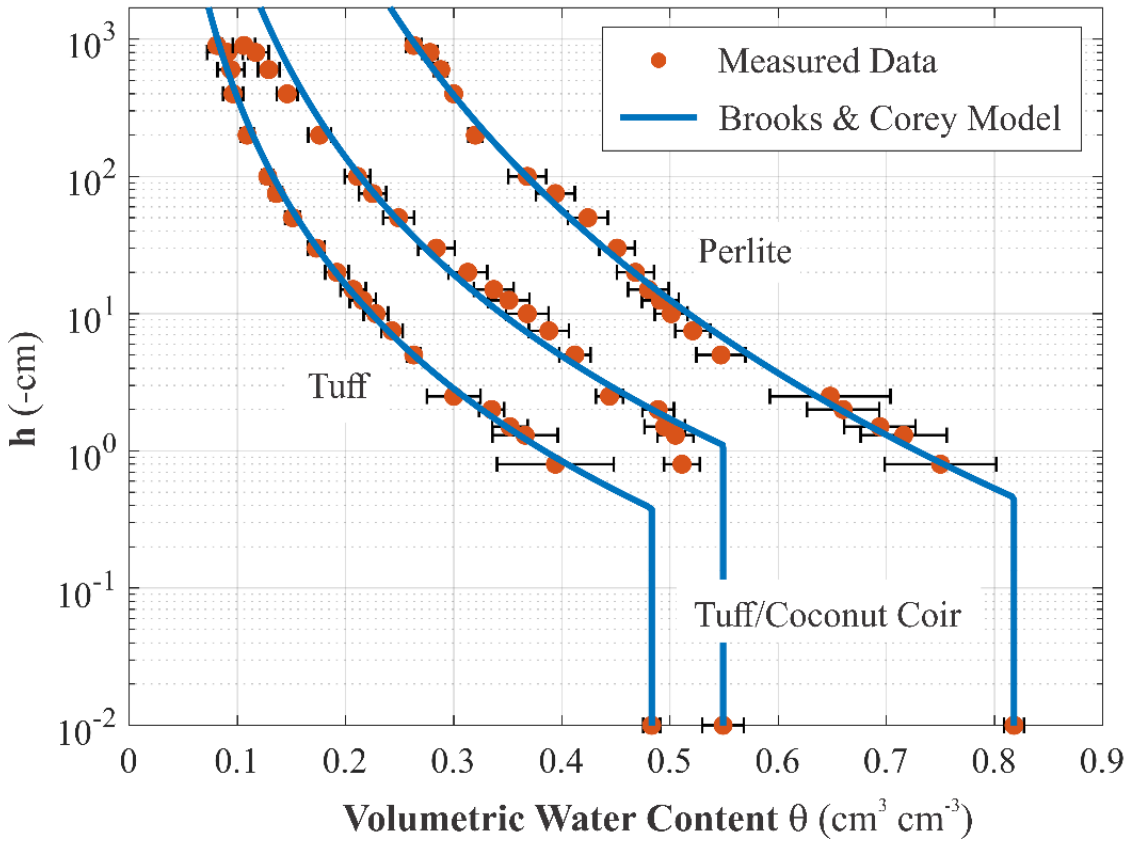

Fig. 2. The *Brooks and Corey* (1964) SWC model fitted to the measured data for tuff (left), the 70/30 vol.-% tuff/coconut coir mixture (center), and perlite (right).

Table. 1. Parameters of the Brooks & Corey SWC model and measured K_{sat}.

Substrate	θ_s ($cm^3\ cm^{-3}$)	θ_r ($cm^3\ cm^{-3}$)	h_b (cm)	λ (-)	K_{sat} ($cm\ h^{-1}$)
Perlite	0.818	0.001	0.458	0.149	305.1
Tuff	0.483	0.014	0.382	0.247	304.2
Tuff/Coconut Coir	0.549	0.014	1.102	0.219	110.7

The phosphorus and ammonium adsorption isotherms were measured with a combination of calorimetric spectrometry and inductively coupled plasma mass spectrometry. Details are provided in *Gohardoust et al.* (2020). The derived isotherm parameters are listed in Table 2.

Table. 2. Nonlinear adsorption isotherm parameters for phosphorus and ammonium.

	Nonlinear Langmuir			
	Phosphorus		Ammonium	
Substrate	k_s ($cm^3\ g^{-1}$)	η ($cm^3\ mg^{-1}$)	k_s ($cm^3\ g^{-1}$)	η ($cm^3\ mg^{-1}$)
Perlite	17.71	984	147.19	3376
Tuff	17.86	66	58.43	135
Tuff/Coconut Coir	24.66	102	42.94	83
	Nonlinear Freundlich			
	k_s ($cm^{3\beta}g^{-\beta}$)	β (-)	k_s ($cm^{3\beta}g^{-\beta}$)	β (-)
Perlite	9.12	0.28	27.66	0.16
Tuff	39.19	0.41	85.23	0.36
Tuff/Coconut Coir	23.10	0.62	72.34	0.43

4. Verification of the Modified ParSWMS Code

Using HYRUS version 3.02.0560 (*Šimůnek et al.*, 2018) as a reference, we meticulously verified the modified ParSWMS code using a simple domain representative of a containerized growth module (Fig. 3). We note that HYDRUS has been extensively tested for specific flow or transport problems based on analytical solutions and measured data (e.g., Cook et al., 2006; Kandelous and Šimůnek, 2010; Karlsson et al., 2015; Lassabatere et al., 2014; Luo and Sophocleous, 2010; Neumann et al., 2011; Vanderborght et al., 2005), and code-to-code validation is an scientifically accepted approach (e.g., Greenwald, 2010; Hardelauf et al., 2007; Kačur and Minár, 2013; Orgogozo et al., 2014).

To mimic irrigation and fertigation with drip emitters, water, phosphorus, ammonium, and nitrate were applied to the surface within two circles with variable flux BCs. When no water or solutes were applied, the variable flux BC was treated as an atmospheric BC to conform with the remaining evaporating substrate surface. The 70/30 vol.-% tuff/coconut coir mixture (see Table 1

for hydraulic parameters), which was dry (i.e., -5000 cm pressure head) at the start of the simulation, was used for validation. While nonlinear Langmuir adsorption was considered for ammonium (NH_4^+) and phosphorus ($H_2PO_4^-$) (see adsorption isotherm parameters in Table 2), nitrate (NO_3^-) did not interact (i.e., adsorb) with the solid phase. The applied concentrations for $H_2PO_4^-$, NH_4^+, and NO_3^- were 20 mg l^{-1}, 20 mg l^{-1}, and 80 mg l^{-1}, respectively. Water was applied every 8 hours at a rate of 1.0 l h^{-1} for 4 minutes with one-fourth allowed to evaporate. The longitudinal and transverse dispersivities were assumed to be 2.0 and 0.2 cm, respectively. The diffusion coefficients of $H_2PO_4^-$, NH_4^+, and NO_3^- in water were assumed as 0.032, 0.063, and 0.068 cm^2 h^{-1}, respectively. Similar values have been reported in Buffle et al. (2007) and Hashitani and Tanaka (1983). The water content tolerance was set to 2×10^{-5} cm^3 cm^{-3}, and the absolute and relative concentration tolerances were set to 10^{-6} mmol cm^{-3} and 10^{-3}, respectively. The flow and transport domains were discretized with finite elements of 1.4-cm in the horizontal direction and 0.7-cm in the vertical direction. Mesh refinements for the top 1-cm layer (i.e., using half of the finite element size) led to a total of 25,264 spatial nodes. The temporal discretization was dynamically calculated throughout the simulation based on the specified initial time step of 2.4E-7 hr. The total simulation time was ten days.

Input files for the modified ParSWMS code were generated with the "Export to ParSWMS" function in HYDRUS. To extract and analyze ParSWMS simulation results, dedicated MATLAB® – Version R2019b (MathWorks, Natick, MA, USA) scripts were created. The simulations were performed on a workstation with two 2.40GHz Intel® Xeon® E5-2630 v3 processors. The hard drive was partitioned with Windows 10 Pro (Version 1909, OS Build 18363.1016) as well as Ubuntu 18.04.5 LTS installed to run the HYDRUS and modified ParSWMS codes, respectively.

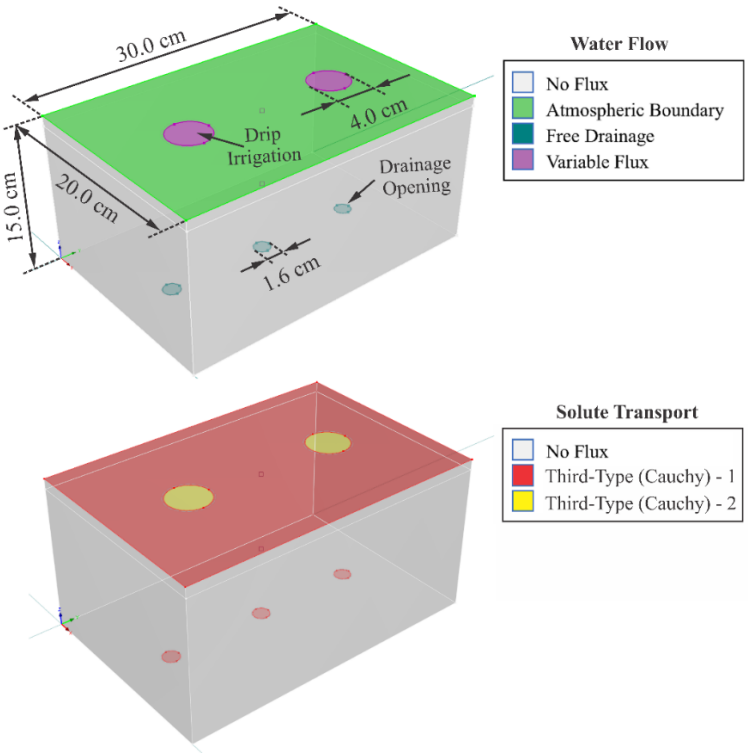

Fig. 3. Flow domain geometry with water flow (top) and solute transport (bottom) BCs. The Cauchy-1 BC is used for the solute concentrations at the top atmospheric and bottom free drainage boundaries. The Cauchy-2 BC is used for the solute concentrations at the drip irrigation boundaries.

The modified ParSWMS and HYDRUS water flow and solute transport simulation results were compared at the end of the simulations based on the normalized root mean squared error (NRMSE) given as:

$$NRMSE = \frac{\sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(C_{PARSWMS_i} - C_{HYDRUS_i}\right)}}{C_{HYDRUS_{max}} - C_{HYDRUS_{min}}} \tag{6}$$

where C is the specific output quantity to be compared, and N is the number of spatial nodes.

The comparison of the modified ParSWMS and HYDRUS volumetric water content (θ) and $H_2PO_4^-$, NH_4^+, and NO_3^- concentration simulation results after 10 days are depicted in Fig. 4. The

coefficients of determination for θ and $H_2PO_4^-$, NH_4^+, and NO_3^- concentrations exceeded 0.999 with associated NRMSEs of less than 1%, which indicates excellent agreement between HYDRUS and the modified ParSWMS code. This is further solidified by the very low mass balance errors of less than 0.3% for water and less than 2.5% for the solutes (Table 3). Notably, the mass balance errors of the modified ParSWMS code are below the errors of HYDRUS.

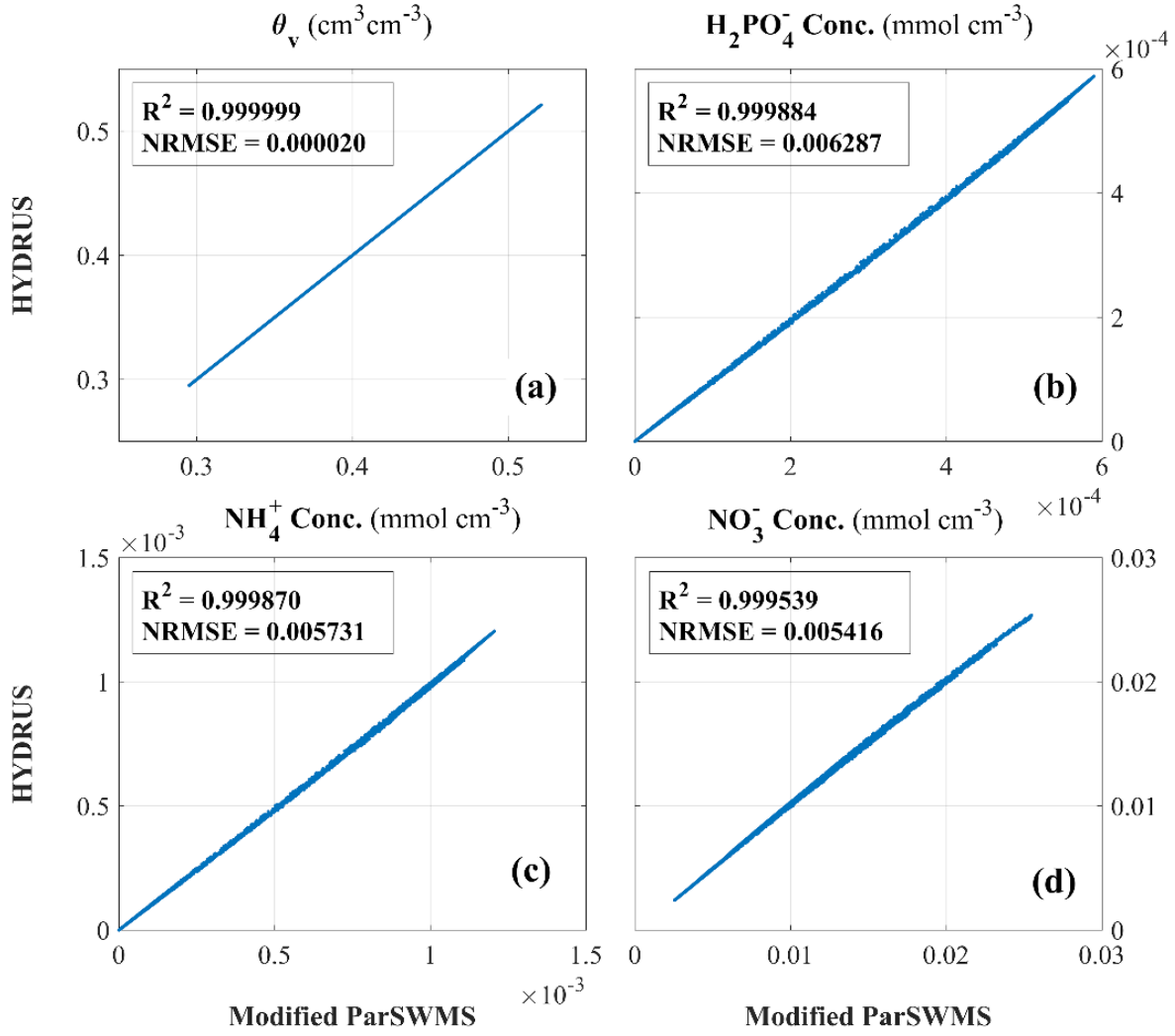

Fig. 4. Comparison of the modified ParSWMS and HYDRUS volumetric water content (θ) (a) and $H_2PO_4^-$ (b), NH_4^+ (c), and NO_3^- (d) concentration simulation results for the 25264 spatial nodes after 10 days.

Table. 3. Cumulative water fluxes and mass balance information.

	Irrigation (cm^3)	E_{ATM}* (cm^3)	E_{VF}† (cm^3)	FD‡ (cm^3)	ΔS_w¶ (cm^3)	Mass Balance Error (%)			
						Water	$H_2PO_4^-$	NH_4^+	NO_3^-
ParSWMS	4001.86	902.8	49.13	912.5	2147.41	0.249	1.806	1.401	0.451
HYDRUS	4001.88	902.7	48.88	912.9	2147.41	0.250	2.421	2.439	0.514

* Water evaporation from areas with an atmospheric BC.
† Water evaporation from areas with a variable flux BC when not irrigated.
‡ Water loss from the free drainage boundaries.
¶ Change in water storage.

5. Evaluation of Preconditioners and Solvers

To simulate water flow and solute transport, separate sets of linear equations need to be solved. To assure computational efficiency, it is imperative to apply the optimal preconditioner/solver combinations, which differ for flow and transport. The discretization of the associated partial differential equations via finite element or finite difference methods yields linear systems of the form:

$$Ax = b \tag{7}$$

with $A \in \mathbb{R}^{N \times N}$ is the coefficient matrix, $b \in \mathbb{R}^N$ is the right-hand side vector, and x is the vector of unknowns.

Depending on the extent of the problem, direct or iterative solvers may be employed to solve Eq. (7). To enhance the numerical robustness and stability of ill-posed problems with very high condition numbers, sparse direct solvers are preferred despite their high memory usage and extended computation times, especially for large three-dimensional problems (*Kwack et al.*, 2016; *Duff and Scott*, 2004; *Benzi*, 2002). On the other hand, sparse iterative solvers such as preconditioned Krylov subspace solvers can most efficiently handle massive linear equation systems (*Dongarra and Sullivan*, 2000). To resolve Eq. (7) iteratively, the solution is approximated

via constructing a sequence of x_n, starting with an initial guess x_0, by moving in an affine subspace $x_0 + \mathcal{K} \subset \mathbb{R}^N$. In order to create suitable subspaces $\mathcal{K}$, methods containing efficient operations such as matrix-vector products (e.g., Krylov subspaces) are of great value (*Del Corso et al.*, 2015). Assuming the residual of the n^{th} iteration $r_n = b - Ax_n$, Krylov subspace projection methods can be separated into three main categories (*Golub and van der Vorst*, 1997): (1) Ritz-Galerkin methods in which x_n is constructed such that the residual is orthogonal to the current subspace – examples are conjugate gradients (CG), the full orthogonalization method (FOM) (*Saad*, 1981), and generalized conjugate gradients (GENCG) (*Eisenstat et al.*, 1983); (2) minimum residual methods in which x_n minimizes the second norm of the residual over $\mathcal{K}^n$ – the generalized minimal residual (GMRES) (*Saad and Schultz*, 1986), the minimum residual (MINRES), ORTHOMIN (*Vinsome*, 1976), and ORTHODIR (*Young and Jea*, 1980) methods fall within this category; and (3) Petrov-Galerkin methods in which x_n is determined so that the residual is orthogonal to some other *n*-dimensional subspace – the biconjugate gradient (Bi-CG) (*Fletcher*, 1975) and quasi-minimal residual (QMR) methods (*Freund and Nachtigal*, 1991) are part of this category. In addition, there are hybrids of the above categories such as the conjugate gradient squared (CGS) (*Sonneveld*, 1989), biconjugate gradient stabilized (Bi-CGSTAB) (*van der Vorst*, 1992), transpose-free quasi-minimal residual (TFQMR) (*Freund*, 1993), and flexible generalized minimal residual (FGMRES) (*Saad*, 1993) methods.

In brief, preconditioning is applied to convert a given problem into a form that is more amenable for numerical solution (*Herbst et al.*, 2008). This means that Eq. (7) is transformed to the form:

$$M_1^{-1} A\, M_2^{-1}\, y = M_1^{-1}\, b, \qquad y = M_2\, x \tag{8}$$

where M_1 and M_2 are nonsingular matrices with inverses that can be rapidly and accurately solved. This leads to faster convergence and requires much less memory (*Simoncini and Szyld*, 2007).

To find optimal preconditioner/solver combinations with regard to both robustness and computational efficiency, we consider a realistic flow and transport simulation applicable for soilless culture growth modules. Note that the optimal preconditioner/solver combinations differ for water flow and solute transport simulations as they exhibit symmetric and asymmetric coefficient matrices, respectively. Hence, they are evaluated separately.

5.1 Preconditioners and Solvers for Water Flow

Computational efficiency is essential when simulating flow and transport processes in complex porous systems, where intricate hydraulic properties and boundary conditions demand high-resolution discretization of the spatial and temporal domains, often leading to several million nodes requiring numerical evaluation. This holds true for soilless plant growth modules, especially when numerous realizations are desired to optimize container geometry, soilless substrate properties, and irrigation and fertigation management.

To evaluate potential preconditioner/solver combinations for water flow, we considered a typical growth module for greenhouse tomato production at the Agricultural Research Organization (ARO) Volcani Center in Israel (Fig. 5a), where water and nutrients are applied via surface drip emitters and the solution drains through circular openings in the bottom of the concrete containers (Fig. 5b). The 70/30 vol.-% tuff/coconut coir mixture with its hydraulic properties listed in Table 1 was used as the substrate for the test case. The applied water flow and solute transport BCs are consistent with the flow domain depicted in Fig. 3. For water flow, this means an atmospheric BC at the top with ten circular inclusions with variable flux BCs to mimic surface drip irrigation (i.e., two angle arrow drippers per tomato plant) and seven openings at the bottom to allow for free

drainage. For root water uptake, a simple root distribution model (*Vrugt et al.*, 2001a, 2001b) was considered for each of the five tomato plants (*Šimůnek et al.*, 2018). While uptake conforms with the *Feddes et al.* (1978) water stress response function, the potential uptake rate was constrained at 13% of the evapotranspiration rate. Nutrients were applied together with irrigation water (see Section 4 for concentrations). Drip irrigation occurred every two hours for 2.75 minutes at a rate of 1.0 liter per hour per angle arrow dripper (i.e., a total of 5.5 liters were applied per day).

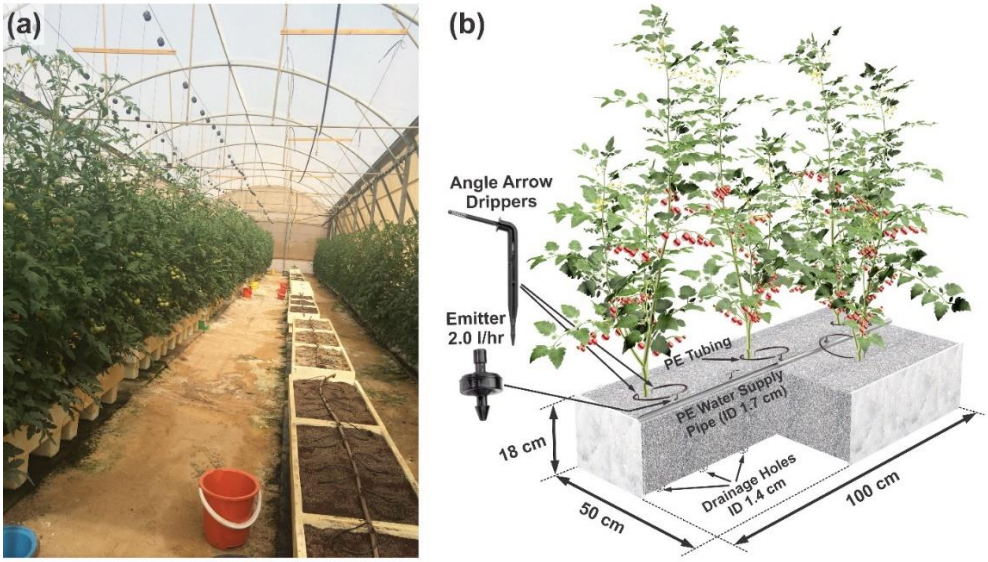

Fig. 5. Greenhouse tomato growth trial at the ARO Volcani Center in Israel (a). Rendering of a concrete growth module used in the greenhouse trial (b).

To consider heterogeneity, independent scaling factors for the pressure head and hydraulic conductivity function were introduced. The scaling factors were generated via sequential Gaussian simulation with the VISIM program (*Hansen and Mosegaard*, 2008) in the open source mGstat MATLAB® toolbox (*Hansen*, 2020) with a variance of 0.05, a mean value of 1.0, and correlation lengths of 4 and 2 cm in the horizontal and vertical directions, respectively. The spatial distribution of the hydraulic conductivity scaling factors within the flow domain is depicted in Fig. 6.

The flow domain was discretized with finite elements of 1.0-cm in the horizontal direction and 0.5-cm in the vertical direction. Mesh refinements for the top 1-cm layer, and in areas with variable flux and free drainage BCs, led to a total of 408,848 nodes and 2,315,763 3D mesh elements.

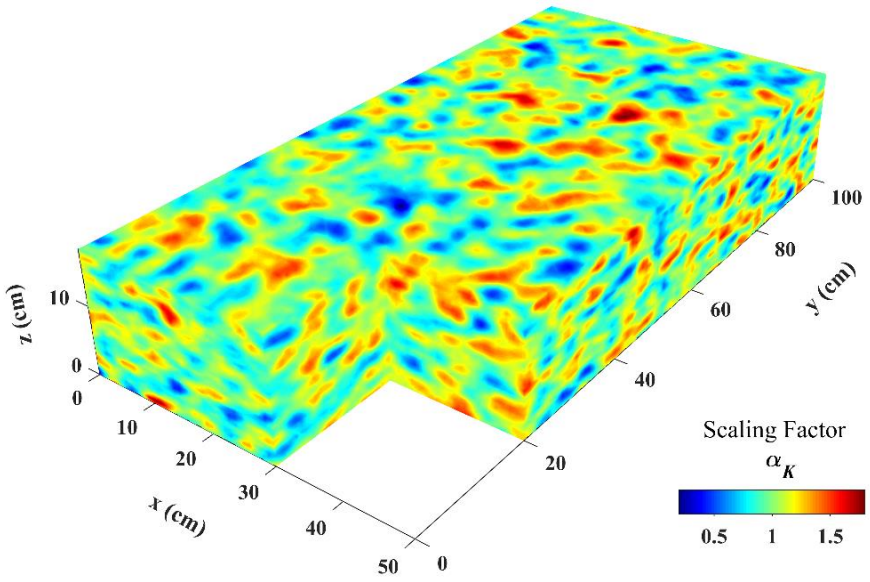

Fig. 6. Spatial distribution of the scaling factor (α_K) for the hydraulic conductivity function within the flow domain.

To solve Eq. (7) with its symmetric sparse coefficient matrix for water flow, a total of 15 preconditioners and 13 solvers were considered (Tab. 4). All preconditioners and solvers, but two, were extracted from the *Portable, Extensible Toolkit for Scientific Computation* (PETSc) library (https://www.mcs.anl.gov/petsc/). Two preconditioners were incorporated from the High Performance Preconditioners (HYPRE) library (http://www.llnl.gov/CASC/hypre/) via the PETSc interface. The sparsity pattern of the coefficient matrix associated with the water flow simulation is depicted in Fig. 7. Because of the large number of considered preconditioner/solver combinations, the total time for each simulation was limited to 10 hours. The simulations were performed on the University of Arizona *Puma* HPC cluster with system specifications provided in Tab. S1. The number of cores (NC) utilized were 12, 24, 48, and 94, respectively.

Table. 4. Preconditioners and solvers considered for the water flow simulations.

Preconditioner	Abbr.	Solver	Abbr.
JACOBI (diagonal scaling)	JACOBI	Conjugate Gradient	CG
Block JACOBI*	BJACOBI	Conjugate Gradient Squared	CGS
Additive Schwarz (Restrict)	ASM-R	Flexible Conjugate Gradient	FCG
Additive Schwarz (Interpolate)	ASM-I	Pipelined FCG	PIPEFCG
Additive Schwarz (Basic)	ASM-B	Generalized Minimal Residual§	GMRES
Additive Schwarz (None)	ASM-N	Deflated GMRES	DGMRES
Shared Blocks ASM (Restrict)	GASM-R	Loose GMRES§	LGMRES
Shared Blocks ASM (Interpolate)	GASM-I	Biconjugate Gradient	BICG
Shared Blocks ASM (Basic)	GASM-B	Biconjugate Gradient Stabilized	BCGS
Shared Blocks ASM (None)	GASM-N	Improved Stabilized BICG	IBCGS
Successive Over Relaxation†	SOR	Enhanced BICG	BCGSL
Symmetric SOR (Eisenstat)	EISEN	Minimum Residual	MINRES
Classical Algebraic Multigrid‡	BOOMERAMG	Chebyshev Iterative Method	CHEBYSHEV
Sparse Approximate Inverse‡	PARASAIL		
Geometric Algebraic Multigrid¶	GAMG		

* Inner preconditioner for each block is ILU(0), i.e., incomplete lower-upper factorization with zero level fill.
† Equivalent to the block Jacobi with SOR on each block.
‡ From the HYPRE library. BOOMERAMG with "HMIS" coarsen type, "ext+I" interpolation type, zero number of levels of aggressive, and 0.5 threshold value for being strongly connected.
¶ GAMG with type 'aggregate' and one smoothing step.
§ With restart parameter 60.

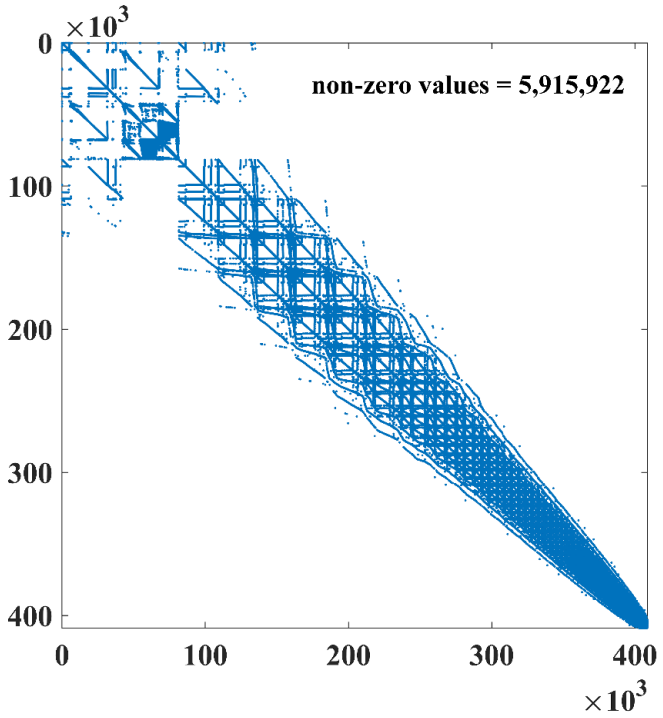

Fig. 7. Sparsity pattern of the symmetric coefficient matrix for water flow simulations.

To compare the computational efficiency of various preconditioner/solver combinations, the wall clock time, which includes preconditioner setup and application time, as well as the time for solver application, was used as a metric. Figure 8 depicts the ten fastest preconditioner/solver combinations together with the ideal speedup line that is defined as the ratio of utilized cores and the number of cores used for the base case (i.e. 12):

$$T_p = \frac{p_{base}}{p} T_{base} \tag{9}$$

where T_p is the simulation time when p processors are utilized, and p_{base} and T_{base} are the number of processors used for the base case and the associated simulation time, respectively. The inset in Fig. 8 shows all successful combinations. As apparent, the choice of the preconditioner/solver combination significantly impacts computational efficiency, as the determined wall clock times vary by more than one order of magnitude. Note that some of the simulations with the GAMG preconditioner did not converge when 94 cores were utilized. The reported wall clock times represent the median of 5 separate simulation runs. The simulation speedup follows the ideal speedup line up to 48 cores, beyond which the performance starts to degrade. This is attributable to the increase of the communication to computation ratio (*Hammond et al.*, 2014). For the executed simulations, the average number of nodes for each core was about 8500 and 4400 when utilizing 48 and 94 cores, respectively. This suggests that for optimal scalability, the minimum number of nodes per processor should fall within this range.

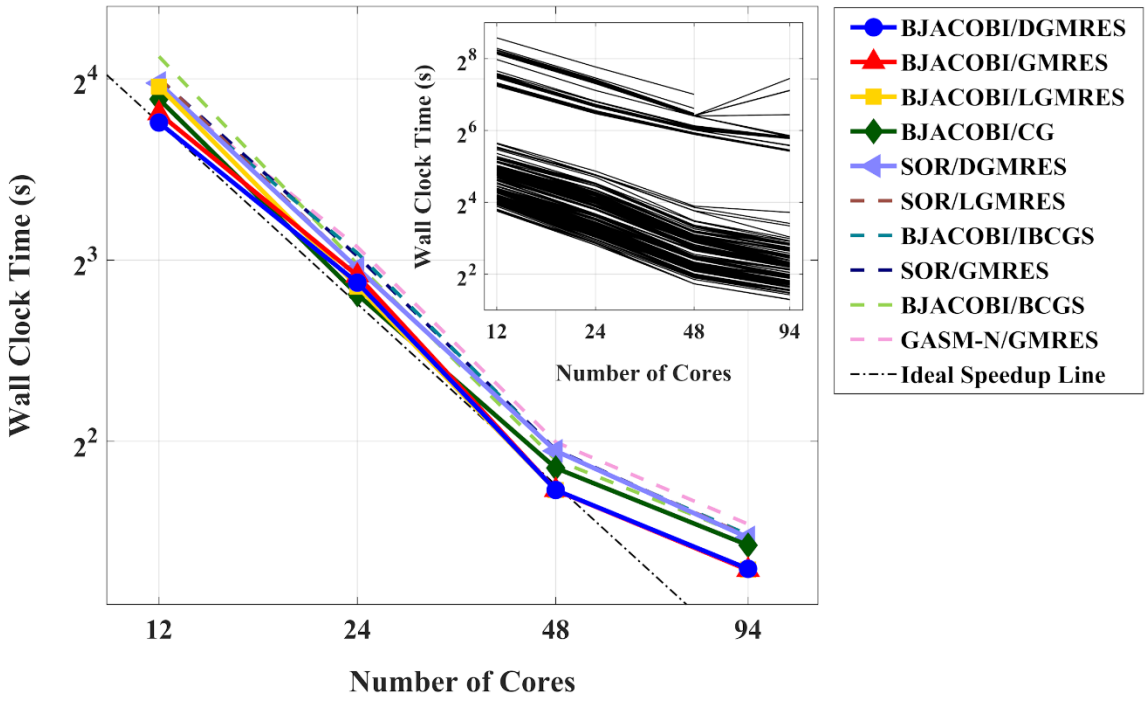

Fig. 8. Wall clock times for the ten most efficient preconditioner/solver combinations for simulation of water flow. The five most efficient combinations are represented by solid lines and symbols, the remaining five by dashed lines. The inset depicts all successful preconditioner/solver combinations.

Because the order of the most efficient combinations is dependent on the number of cores (NC), the wall clock times were normalized by the smallest value for each NC group (i.e., 12, 24, 48, and 94) and then averaged. The five fastest combinations and their associated normalized times are listed in Table 5. Although the DGMRES solver applied in conjunction with the BJACOBI preconditioner performed better than other widely used solvers such as CG, the differences are minor between the five computationally most efficient combinations (Table 5). This leads to the conclusion that BJACOBI, which performs the incomplete lower-upper factorization in each block (with zero level fill), is the preconditioner of choice for our water flow simulations. This finding in general concurs with *Herbst et al.* (2008), who investigated the performance of five

preconditioners in combination with the CG solver for simulation of one- and three-dimensional water flow (i.e., Richards equation) in porous media. In contrast to their results, the algebraic multigrid preconditioner (GAMG) did not work well for our simulations, which is most likely because of the more complex water distributions within our flow domain due to the implementation of surface drip irrigation, root water uptake, and free drainage from the bottom boundary.

Table. 5. The most efficient preconditioner/solver combinations for the water flow simulations.

Rank	Preconditioner/Solver Combination	Normalized Time
1	BJACOBI/DGMRES	1.0128
2	BJACOBI/GMRES	1.0276
3	BJACOBI/LGMRES	1.0450
4	BJACOBI/CG	1.0704
5	SOR/DGMRES	1.1407

5.2 Preconditioners and Solvers for Solute Transport

The test case for solute transport is identical to the test case in *Section 5.1* with regard to water flow. In addition, $H_2PO_4^-$, NH_4^+, and NO_3^- were applied with the irrigation water at concentrations of 20 mg l^{-1}, 20 mg l^{-1}, and 80 mg l^{-1}, respectively. While nonlinear Langmuir adsorption was considered for ammonium (NH_4^+) and phosphorus ($H_2PO_4^-$), nitrate (NO_3^-) did not interact with the solid phase. Fifteen preconditioners and 12 solvers were evaluated (Table 6).

Table. 6. Preconditioners and solvers considered for the solute transport simulations.

Preconditioner	Abbr.	Solver	Abbr.
JACOBI (diagonal scaling)	JACOBI	Transpose Free QMR[§]	TFQMR
Block JACOBI[*]	BJACOBI	A Variant of QMR	TCQMR
Additive Schwarz (Restrict)	ASM-R	Generalized Minimal Residual[‖]	GMRES
Additive Schwarz (Interpolate)	ASM-I	Flexible GMRES[‖]	FGMRES
Additive Schwarz (Basic)	ASM-B	Deflated GMRES	DGMRES
Additive Schwarz (None)	ASM-N	Pipelined GMRES	PGMRES
Shared Blocks ASM (Restrict)	GASM-R	Pipelined Flexible GMRES	PIPEGMRES
Shared Blocks ASM (Interpolate)	GASM-I	LGMRES[‖]	LGMRES
Shared Blocks ASM (Basic)	GASM-B	Conjugate Gradient Squared	CGS
Shared Blocks ASM (None)	GASM-N	Stabilized Biconjugate Gradient	BCGS
Successive Over Relaxation[†]	SOR	Enhanced BICG	BCGSL
Symmetric SOR (Eisenstat)	EISEN	Improved Stabilized BICG[‖]	IBCGS
Classical Algebraic Multigrid[‡]	BOOMERAMG		
Sparse Approximate Inverse[‡]	PARASAIL		
Geometric Algebraic Multigrid[¶]	GAMG		

[*] Inner preconditioner for each block is ILU(0), i.e., incomplete lower-upper factorization with zero level fill.
[†] Equivalent to BJACOBI with SOR on each block.
[‡] From the HYPRE library. The BOOMERAMG with "HMIS" coarsen type, "ext+i" interpolation type, zero number of levels of aggressive, and 0.5 threshold value for being strongly connected.
[¶] GAMG with type 'aggregate' and zero smoothing steps.
[§] QMR: Quasi Minimal Residual method.
[‖] With restart parameter 60.

The ten computationally most efficient preconditioner/solver combinations for solute transport are depicted in Fig. 9 together with the ideal speedup line. The inset shows all successful combinations, which vary over two orders of magnitude. As for water flow, the simulation speedup follows the ideal speedup line up to 48 cores and then the performance starts degrading. Combinations with the GAMG preconditioner were the least efficient. Although the GAMG preconditioner performed better for solute transport, it is not among the top-ranked. These results differ from *Sbai and Larabi* (2020), who simulated field scale chemical transport in groundwater. The deviations can be attributed to the consideration of adsorbing solutes in our simulations as well as the generally better performance of the GAMG preconditioner for large transport domains (*Sbai and Larabi*, 2020).

The most efficient combinations consisted of preconditioners of the successive over relaxation category (i.e., SOR and EISENSTAT) and variations of the GMRES solver. It should be noted that the parallel SOR and EISENSTAT preconditioners are equivalent to BJACOBI with SOR on each block and are therefore not true parallel preconditioners. As evident from Fig. 9 and Table 7, the SOR/FGMRES combination is best suited for our application.

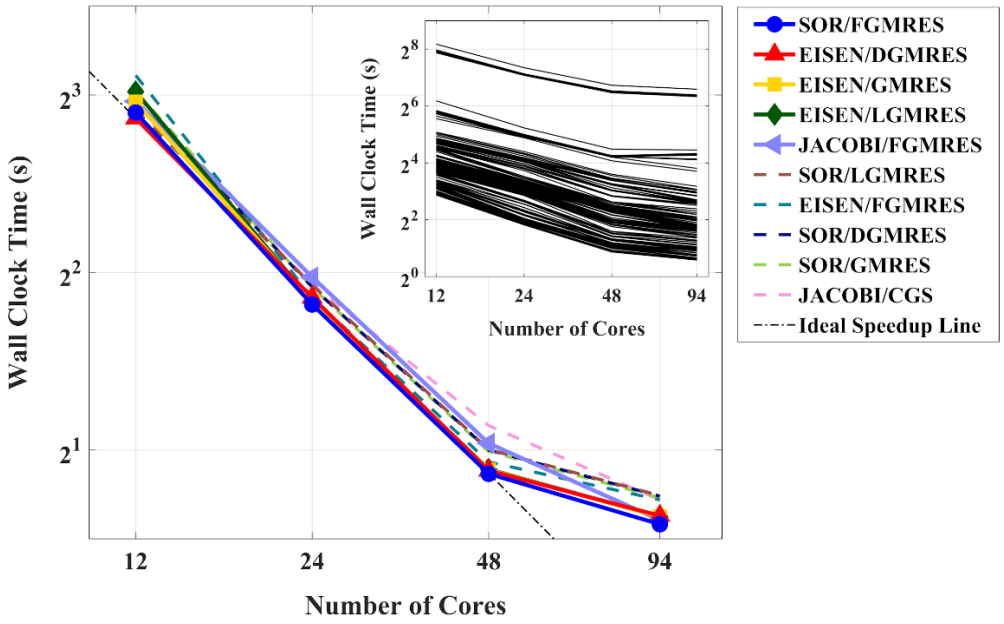

Fig. 9. Wall clock times for the ten most efficient preconditioner/solver combinations for simulation of solute transport. The five most efficient combinations are represented by solid lines and symbols, the remaining five by dashed lines. The inset depicts all successful preconditioner/solver combinations.

Table. 7. The most efficient preconditioner/solver combinations for the solute transport.

Ranking	Preconditioner/Solver Combination	Normalized Time
1	SOR/FGMRES	1.0056
2	EISENSTAT/DGMRES	1.0186
3	EISENSTAT/GMRES	1.0312
4	EISENSTAT/LGMRES	1.0421
5	JACOBI/FGMRES	1.0810

6. Computational Efficiency and Stability of HYDRUS and the Modified ParSWMS

As already indicated in *Section 5*, computational efficiency and stability are essential when simulating flow and transport processes in soilless plant growth modules, especially when a vast number of realizations is required. Hence, it is of interest to compare the Windows OS based HYDRUS with the Linux OS based modified ParSWMS. Because workstations and HPC clusters are not intercomparable, such comparison is only viable on a workstation, with both Windows (i.e., Windows 10 Pro Version 1909, OS Build 18363.1016) and Linux (i.e., Ubuntu 18.04.5 LTS) installed on a partitioned hard drive. A workstation with two 2.40GHz Intel® Xeon® E5-2630 v3 Dual 8-Core processors and 32GB RAM was used for the comparison.

HYDRUS benefits from a powerful GUI with numerous advanced features such as the ability to create or import complex domain geometries or to specify various spatially variable properties such as materials, initial conditions, boundary conditions, and domain properties either directly on the finite element mesh or on geometric objects independent of the mesh (*Šimůnek et al.*, 2016). To utilize parallel computing and the computing prowess of the latest multicore workstations, HYDRUS relies on the HYdrus PARallelized (HYPAR) module and the Microsoft Parallel Patterns Library to numerically solve Eq. (7) for water flow and solute transport with the CG and ORTHOMIN solvers, respectively. Equation (7) is preconditioned with the incomplete lower upper factorization method. Both the solvers and the preconditioner are from the ORTHOFEM library (*Mendoza et al.*, 1991).

The test case for the stability and efficiency comparison conforms with the descriptions in *Sections 5.1* and *5.2*, except that the SWC and hydraulic conductivity functions were not scaled for the considered tuff, tuff/coconut coir, and perlite substrates (see hydraulic properties in Tab. 1). The simulation duration was 14 days (336 hrs). The flow domain was discretized with finite elements

of 1.90-cm in the horizontal direction and 0.95-cm in the vertical direction. Mesh refinements for the top 1-cm layer and in areas with variable flux and free drainage BCs, led to a total of 132,570 nodes and 726,264 3D mesh elements. Note that the hydraulic characteristics are calculated directly from the hydraulic functions by setting the lower and upper limits of the internal interpolation tables to zero in both HYDRUS and the modified ParSWMS code. The BJACOBI/DGMRES and SOR/FGMRES preconditioner/solver combinations were used for ParSWMS water flow and solute transport simulations, respectively.

Figure 10 shows the temporal evolution of mass balance errors for water flow and $H_2PO_4^-$, NH_4^+, and NO_3^- transport. The water mass balance errors are below 0.15% for all substrates for both the HYDRUS and ParSWMS simulations, which we consider quite accurate. The mass balance errors for solute transport are higher for both HYDRUS and the modified ParSWMS. An extreme error of about 80% for NO_3^- is evident for the HYDRUS simulation for tuff, which can be attributed to simulation instability issues starting at around 48 hrs. This is likely due to very localized increases in NO_3^- concentrations at the atmospheric BC because of evaporation (Fig. 11, bottom right). In comparison, the maximum mass balance error for NO_3^- and tuff simulated with the modified ParSWMS is 0.54%. Note that the instability issues experienced with HYDRUS can be potentially resolved by increasing the density of the 3D mesh close to the surface at the cost of computational efficiency. Notably, the ParSWMS solute mass balance errors are significantly lower than the HYDRUS errors.

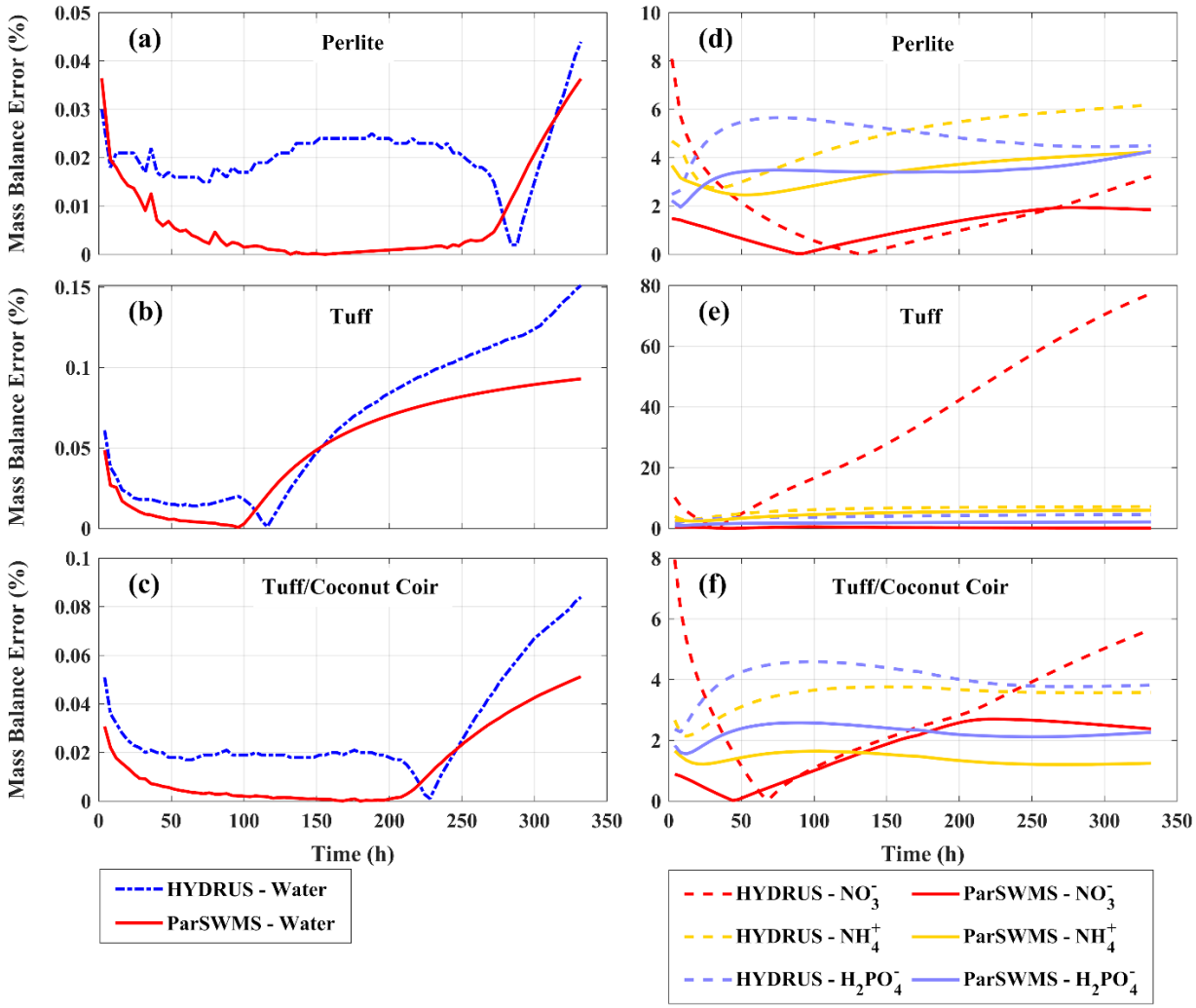

Fig. 10. Temporal mass balance error evolutions for water flow (left) and $H_2PO_4^-$, NH_4^+, and NO_3^- transport (right) simulated with both HYDRUS and the modified ParSWMS code.

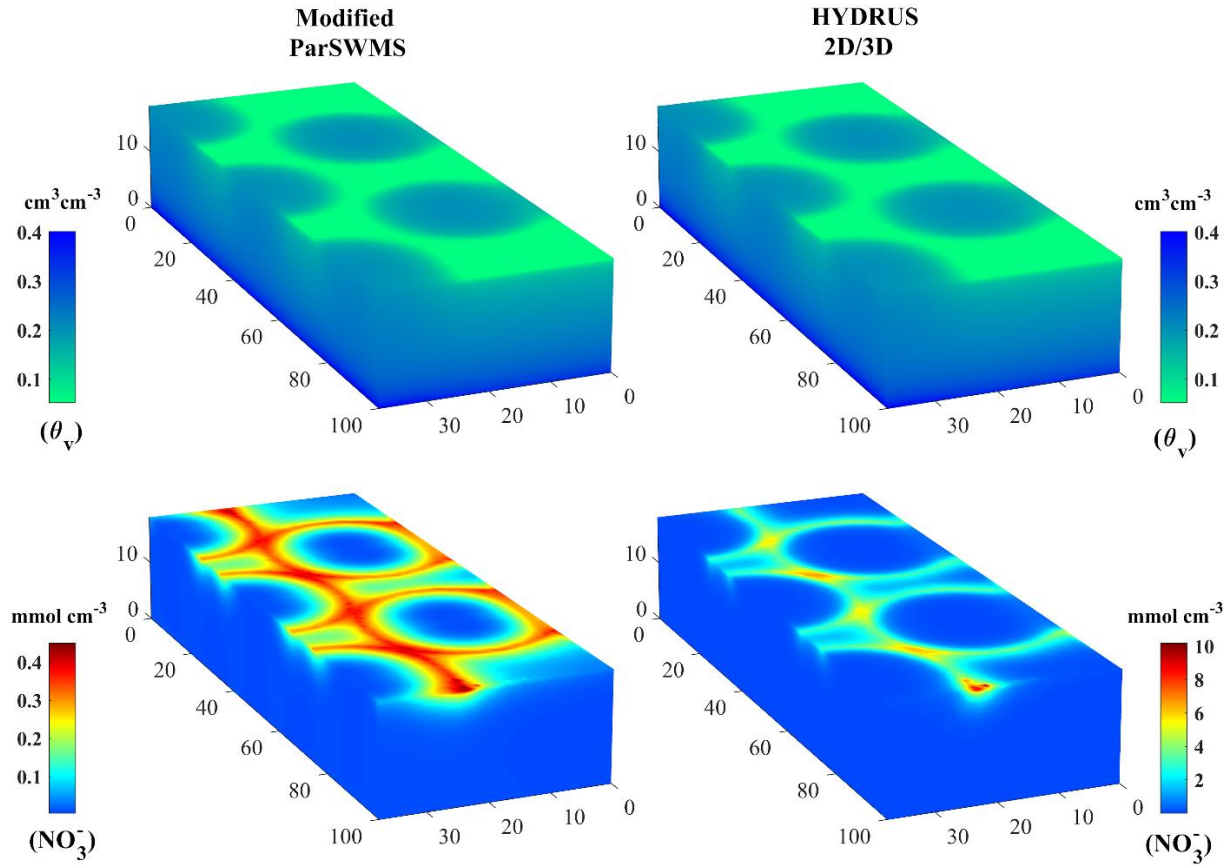

Fig. 11. Water content and NO_3^- concentration distributions at the end of the HYDRUS and modified ParSWMS simulations. Note the localized very high NO_3^- concentrations depicted in the bottom right graph for HYDRUS.

Figure 12 shows a comparison of computational efficiencies of the HYDRUS and modified ParSWMS codes. Besides the simulations performed on the workstation, we ran ParSWMS on the *Ocelote* HPC (28 cores) and the recently installed *Puma* HPC (94 cores) clusters (for specifications see Table S1) to demonstrate the massive increase in scalability when HPC environments are utilized. The simulation speedup expressed in terms of normalized wall clock times (see values on top of the bars in Fig. 12), of the modified ParSWMS code is about 22% faster than that of HYDRUS on the workstation, and about 94% faster on the Puma HPC cluster with 94 cores. Note that while an increase in the number of processing cores generally speeds up the simulations,

beyond a certain core threshold, the computational efficiency starts to decline due to an increase of the communication to computation ratio.

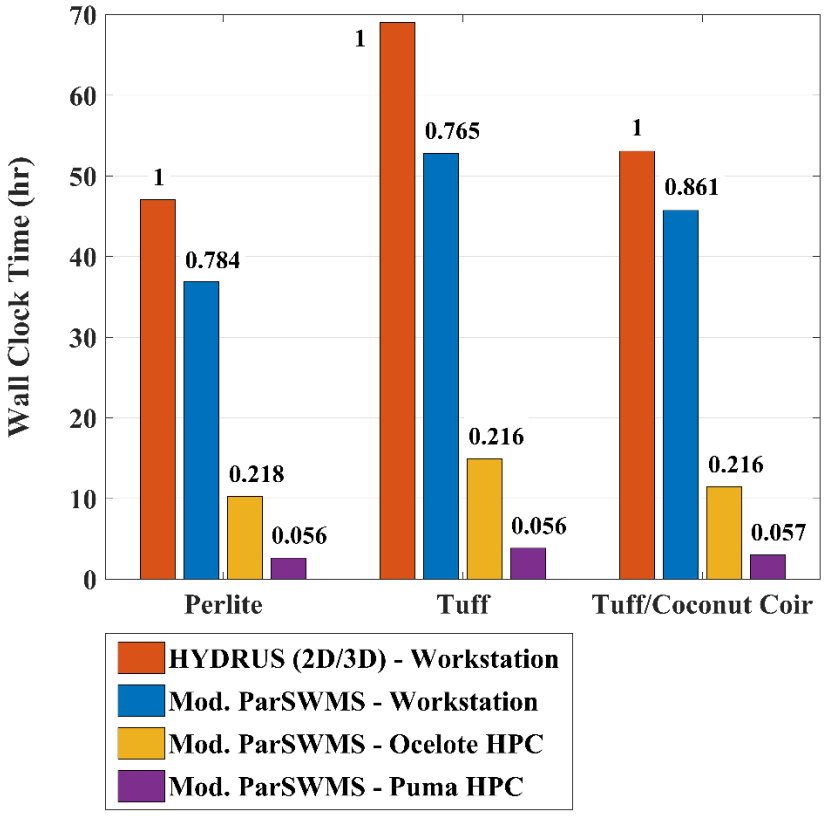

Fig. 12. Comparison of computational efficiencies of the HYDRUS and the modified ParSWMS codes for perlite (left), tuff (center), and tuff/coconut coir (right). The values on top of the bars represent the clock times normalized with the HYDRUS clock times.

Conclusions

With the goal of performing a vast number of flow and transport simulations on HPC clusters to aid the design and management of soilless culture growth modules, we modified the open source 3D ParSWMS code to not only enable nonlinear equilibrium solute adsorption and multi-solute transport simulations, but also the application of boundary conditions to realistically represent typical soilless culture systems. The modified ParSWMS code was thoroughly tested using

HYDRUS as a reference. To optimize the computational efficiency of the modified ParSWMS, numerous preconditioner/numerical solver combinations were tested for both the water flow and solute transport equations. While the BJACOBI/DGMRES preconditioner/solver combination was the most efficient for water flow, SOR/FGMRES worked best for solute transport. In general, GMRES-type and CG solvers applied in conjunction with the BJACOBI preconditioner were very efficient for water flow simulations. The most efficient combinations for solute transport consisted of preconditioners of the successive over relaxation category (i.e., SOR and EISENSTAT) and variations of the GMRES solver. To compare the stability and computational efficiency of modified ParSWMS with HYDRUS, a real growth module from a production scale greenhouse experiment at the ARO Volcani Center in Israel was simulated considering three soilless substrates and $H_2PO_4^-$, NH_4^+, and NO_3^- transport. The simulations were performed on a workstation and two HPC clusters. The results revealed that the modified ParSWMS was about 22% more efficient than HYDRUSwhen the simulations on the workstation were compared. Simulations on the HPC clusters were up to 94% more efficient when 94 cores were utilized. If the full computational power of the HPC clusters (i.e., no competing projects and up to 23616 available cores) was to be utilized, simulation times would again be drastically reduced to a very small fraction of the HYDRUS wall clock time. While we experienced some stability issues for tuff and NO_3^- with HYDRUS, leading to an unrealistically high mass balance error, all modified ParSWMS simulations were stable on both the workstation and HPC clusters. In general, all water and solute mass balance errors generated with the modified ParSWMS where bellow the mass balance errors of HYDRUS.

The availability of the modified and tested ParSWMS code opens new avenues for streamlining 3D water flow and solute transport simulations for complex porous media via the utilization of

HPC environments. Simulations that took days to complete can now be resolved within a matter of a few hours, allowing for a vast number of realizations within a short period of time.

Availability of Software

All computer codes and Matlab scripts created for this project are available from the corresponding author upon reasonable request.

Acknowledgments

This research was funded by The United States - Israel Binational Agricultural Research and Development Fund (BARD), grant number US-4764-14 R, and by the United States Department of Agriculture (USDA) National Institute of Food and Agriculture (NIFA), Hatch/Multi-State project number ARZT-1370600-R21-189. We thank Dr. Asher Bar-Tal from the ARO Volcani Center in Israel for providing details about the design and management of the tomato growth experiment that was simulated in this study.

Author Statement

Mohammad R. Gohardoust: Conceptualization, Methodology, Software, Formal analysis, Investigation, Writing - Original Draft, Writing - Review & Editing

Jirka Šimůnek: Conceptualization, Methodology, Writing - Review & Editing

Horst Hardelauf: Conceptualization, Software, Writing - Review & Editing

Markus Tuller: Conceptualization, Methodology, Formal analysis, Writing - Original Draft, Writing - Review & Editing, Supervision, Funding acquisition

References

Abad, M., Fornes, F., Carrión, C., Noguera, V., Noguera, P., Maquieira, Á., & Puchades, R. (2005). Physical Properties of Various Coconut Coir Dusts Compared to Peat. *HortScience*, 40(7), 2138–2144.

Bar-Tal, A., Saha, U.K., Raviv, M., & Tuller, M. (2019). Inorganic and Synthetic Organic Components of Soilless Culture and Potting Mixtures. In M. Raviv, H.J. Lieth, & A. Bar-Tal (Eds.), Soilless Culture – Theory and Practice, 2nd Ed. (pp. 264–267). London: Academic Press (Elsevier).

Benzi, M. (2002). Preconditioning Techniques for Large Linear Systems: A Survey. *J. Comput. Phys.*, 182(2), 418–477. https://doi.org/10.1006/jcph.2002.7176.

Brooks, R.H., & Corey, A.T. (1966). Properties of Porous Media Affecting Fluid Flow. *J. Irrig. Drain. Div., Am. Soc. Civ. Eng.*, 92(2), 61–88.

Buffle, J., Zhang, Z., & Startchev, K., (2007). Metal Flux and Dynamic Speciation at (Bio)interfaces. Part I: Critical Evaluation and Compilation of Physicochemical Parameters for Complexes with Simple Ligands and Fulvic/Humic Substances. *Environ. Sci. Technol.*, 41, 7609–7620. https://doi.org/10.1021/es070702p.

Cook, F.J., Fitch, P., Thorburn, P.J., Charlesworth, P.B., & Bristow, K.L. (2006). Modelling trickle irrigation: Comparison of analytical and numerical models for estimation of wetting front position with time. *Environ. Model. Softw*., 21, 1353–1359. https://doi.org/10.1016/j.envsoft.2005.04.018.

Del Corso, G.M., Menchi, O., & Romani, F. (2015). Krylov Subspace Methods for Solving Linear Systems. Technical Report, University of Pisa, Department of Computer Science, Pisa, Italy.

Dongarra, J., & Sullivan, F. (2000). Guest Editors' Introduction: The Top 10 Algorithms. *Comp. Sci. Eng.*, 2(1), 22–23. https://doi.org/10.1109/MCISE.2000.814652 .

Duff, I.S., & Scott, J.A. (2004). A Parallel Direct Solver for Large Sparse Highly Unsymmetric Linear Systems. *ACM T. Math. Software*, 30(2), 95–117. https://doi.org/10.1145/992200.992201

Eisenstat, S.C., Elman, H.C., & Schultz, M.H. (1983). Variational Iterative Methods for Nonsymmetric Systems of Linear Equations. *SIAM J. Numer. Anal.*, 20(2), 345–357.

Evans, M.R., Konduru, S., & Stamps, R.H. (1996). Source Variation in Physical and Chemical Properties of Coconut Coir Dust. *HortScience*, 31(6), 965–967.

Feddes, R.A., Kowalik, P.J., & Zaradny, H. (1978). Simulation of Field Water Use and Crop Yield. New York: John Wiley & Sons.

Fletcher, R. (1975). Conjugate Gradient Methods for Indefinite Systems. In G.A. Watson (Ed.), Numerical Analysis (pp. 73–89). Berlin: Springer Verlag.

Freund, R.W., & Nachtigal, N.M. (1991). QMR: A Quasi-Minimal Residual Method for Non-Hermitian Linear Systems. *Numer. Math*, 60, 315–339.

Freund, R.W. (1993). A Transpose-Free Quasi-Minimal Residual Algorithm for Non-Hermitian Linear Systems. *SIAM J. Sci. Comput.*, 14(2), 470–482. https://doi.org/10.1137/0914029.

Gohardoust, M.R., Bar-Tal, A., Effati, M., & Tuller, M. (2020). Characterization of Physicochemical and Hydraulic Properties of Organic and Mineral Soilless Culture Substrates and Mixtures. *Agronomy*, 10(9), 1403. https://doi.org/10.3390/agronomy10091403.

Golub, G.H., & van der Vorst, H.A. (1997). Closer to the Solutions: Iterative Linear Solvers. In I.S. Duff, & G.A. Watson (Eds.), The State of the Art in Numerical Analysis, (pp. 63–92). Oxford: Clarendon Press.

Greenwald, M. (2010). Verification and validation for magnetic fusion. *Phys. Plasmas*, 17, 058101. https://doi.org/10.1063/1.3298884.

Hammond, G.E., Lichtner, P.C., & Mills, R.T. (2014). Evaluating the performance of parallel subsurface simulators: An illustrative example with PFLOTRAN. *Water Resour. Res.*, 50, 208–228. https://doi.org/10.1002/2012WR013483.

Hansen, T.M., & Mosegaard, K. (2008). VISIM: Sequential Simulation for Linear Inverse Problems. *Comput. Geosci.*, 34(1), 53–76. https://doi.org/10.1016/j.cageo.2007.02.003.

Hansen, T.M. (2020). mGstat: A Geostatistical Matlab toolbox. http://mgstat.sourceforge.net/ Accessed 4 September 2020.

Hardelauf, H., Javaux, M., Herbst, M., Gottschalk, S., Kasteel, R., Vanderborght, J., & Vereecken, H. (2007). PARSWMS: A Parallelized Model for Simulating Three-Dimensional Water Flow and Solute Transport in Variably Saturated Soils. *Vadose Zone J.*, 6(2), 255–259. https://doi.org/10.2136/vzj2006.0156.

Hashitani, T., & Tanaka, K. (1983). ChemInform Abstract: measurements of self-diffusion coefficients of the nitrate ion in aqueous solutions of potassium nitrate and calcium nitrate. *Chem. Informationsd.*, 14. https://doi.org/10.1002/chin.198346019.

Herbst, M., Gottschalk, S., Reißel, M., Hardelauf, H., Kasteel, R., Javaux, M., Vanderborght, J., & Vereecken, H. (2008). On preconditioning for a parallel solution of the Richards equation. Comput Geosci., 34(12), 1958–1963. https://doi.org/10.1016/j.cageo.2008.02.020.

Kačur, J., & Minár, J. (2013). A Benchmark Solution for Infiltration and Adsorption of Polluted Water Into Unsaturated–Saturated Porous Media. *Transp. Porous Media*, 97, 223–239. https://doi.org/10.1007/s11242-012-0119-5.

Karlsson, S.C., Langergraber, G., Pell, M., Dalahmeh, S., Vinnerås, B., & Jönsson, H. (2015). Simulation and verification of hydraulic properties and organic matter degradation in sand filters for greywater treatment. *Water Sci. Technol*., 71, 426–433. https://doi.org/10.2166/wst.2015.003.

Kandelous, M.M., & Šimůnek, J. (2010). Comparison of numerical, analytical, and empirical models to estimate wetting patterns for surface and subsurface drip irrigation. *Irrig. Sci.*, 28, 435–444. https://doi.org/10.1007/s00271-009-0205-9.

Kwack, JH., Bauer, G.H., & Koric, S. (2016, May). Performance Test of Parallel Linear Equation Solvers on Blue Waters – Cray XE6/XK7 system. Conference paper, Proceedings of the Cray User Group Meeting (CUG 2016), London, England.

Lassabatere, L., Yilmaz, D., Peyrard, X., Peyneau, P.E., Lenoir, T., Šimůnek, J., Angulo-Jaramillo, R. (2014). New Analytical Model for Cumulative Infiltration into Dual-Permeability Soils. *Vadose Zone J.*, 13. https://doi.org/10.2136/vzj2013.10.0181.

Luo, Y., & Sophocleous, M. (2010). Seasonal groundwater contribution to crop-water use assessed with lysimeter observations and model simulations. *J. Hydrol*., 389, 325–335. https://doi.org/10.1016/j.jhydrol.2010.06.011.

Mendoza, C.A., Therrien, R., & Sudicky, E.A. (1991). ORTHOFEM User's Guide, Version 1.02. Waterloo, Ontario, Canada: Waterloo Centre for Groundwater Research, University of Waterloo.

Noland, D.A., Spomer, L.A., & Williams, D.J. (1992). Evaluation of Pumice as a Perlite Substitute for Container Soil Physical Amendment. *Commun. Soil Sci. Plan.*, 23(13-14), 1533-1547. https://doi.org/10.1080/00103629209368685.

Neuman, S. P., Feddes, R. A., & Bresler, E. (1974). Finite element simulation of flow in saturated-unsaturated soils considering water uptake by plants, Third Annual Report, Project No. A10-SWC-77, Hydraulic Engineering Lab., Technion, Haifa, Israel.

Neumann, L.E., Šimůnek, J., & Cook, F.J. (2011). Implementation of quadratic upstream interpolation schemes for solute transport into HYDRUS-1D. *Environ. Model. Softw*., 26, 1298–1308. https://doi.org/10.1016/j.envsoft.2011.05.010.

Orgogozo, L., Renon, N., Soulaine, C., Hénon, F., Tomer, S.K., Labat, D., Pokrovsky, O.S., Sekhar, M., Ababou, R., & Quintard, M. (2014). An open source massively parallel solver for Richards equation: Mechanistic modelling of water fluxes at the watershed scale. *Comput. Phys. Commun.*, 185, 3358–3371. https://doi.org/10.1016/j.cpc.2014.08.004.

Raviv, M., Lieth, H.J., & Bar-Tal, A. (2019). Significance of Soilless Culture in Agriculture. In M. Raviv, H.J. Lieth, & A. Bar-Tal (Eds.), Soilless Culture – Theory and Practice, 2nd Ed. (pp. 3–14). London: Academic Press (Elsevier).

Saad, Y. and Schultz, M.H., 1986. GMRES: A Generalized Minimal Residual Algorithm for Solving Nonsymmetric Linear Systems. *SIAM J. Sci. Stat. Comput.*, 7(3), 856–869.

Saad, Y. (1981). Krylov Subspace Methods for Solving Large Unsymmetric Linear Systems. *Math. Comput.*, 37(155), 105–126.

Saad, Y. (1993). A Flexible Inner-Outer Preconditioned GMRES Algorithm. *SIAM J. Sci. Comput.*, 14(2), 461–469. https://doi.org/10.1137/0914028.

Savvas, D. (2003). Hydroponics: A modern technology supporting the application of integrated crop management in greenhouse. *J. Food Agri. Environ*., 1, 80–86.

Savvas, D., & Gruda, N. (2018). Application of soilless culture technologies in the modern greenhouse industry – A review. *Eur. J. Hortic. Sci*., 83, 280–293. https://doi.org/10.17660/eJHS.2018/83.5.2

Sbai, M.A., & Larabi, A. (2020). On Solving Groundwater Flow and Transport Models with Algebraic Multigrid Preconditioning. *Groundwater*. https://doi.org/10.1111/gwat.13016.

Silber, A., & Raviv, M. (1996). Effects on Chemical Surface Properties of Tuff by Growing Rose Plants. *Plant Soil*, 186, 353–360.

Silber, A., Bar-Yosef, B., & Chen, Y. (1999). pH-Dependent Kinetics of Tuff Dissolution. *Geoderma*, 93(1-2), 125–140. https://doi.org/10.1016/S0016-7061(99)00048-8.

Silber, A., Bar-Yosef, B., Singer, A., & Chen, Y. (1994). Mineralogical and Chemical Composition of Three Tuffs from Northern Israel. *Geoderma*, 63(2), 123–144. https://doi.org/10.1016/0016-7061(94)90002-7.

Simoncini, V., & Szyld, D.B. (2007). Recent Computational Developments in Krylov Subspace Methods for Linear Systems. *Numer. Linear Algebra Appl.*, 14, 1–59.

Šimůnek, J., Huang, K., & van Genuchten, M.Th. (1995). The SWMS-3D Code for Simulating Water Flow and Solute Transport in Three-Dimensional Variably-Saturated Media. Riverside: USDA ARS U.S. Salinity Laboratory, (Research Report No. 139).

Šimůnek, J., van Genuchten, M.Th., & Šejna, M. (2016). Recent Developments and Applications of the HYDRUS Computer Software Packages. *Vadose Zone J.*, 15(7). https://doi.org/10.2136/vzj2016.04.0033

Šimůnek, J., M. Šejna, and M. Th. van Genuchten, (2018). New features of version 3 of the HYDRUS (2D/3D) computer software package, *J. Hydrol. Hydromech.*, 66(2), 133-142. https://doi.org/10.1515/johh-2017-0050

Sonneveld, P. (1989). CGS, A Fast Lanczos-Type Solver for Nonsymmetric Linear Systems. *SIAM J. Sci. and Stat. Comput.*, 10(1), 36–52. https://doi.org/10.1137/0910004.

van Genuchten, M.Th. (1980). A Closed-form Equation for Predicting the Hydraulic Conductivity of Unsaturated Soils. *Soil Sci. Soc. Am. J.*, 44, 892–898.

van der Vorst, H.A. (1992). Bi-CGSTAB: A Fast and Smoothly Converging Variant of Bi-CG for the Solution of Nonsymmetric Linear Systems. *SIAM J. Sci. and Stat. Comput.*, 13(2), 631–644. https://doi.org/10.1137/0913035.

Vanderborght, J., Kasteel, R., Herbst, M., Javaux, M., Thiery, D., Vanclooster, M., Mouvet, C., & Vereecken, H. (2005). A Set of Analytical Benchmarks to Test Numerical Models of Flow and Transport in Soils. *Vadose Zone J.*, 4, 206–221. https://doi.org/10.2113/4.1.206.

Vinsome, P.K.W. (1976, February). Orthomin, an Iterative Method for Solving Sparse Sets of Simultaneous Linear Equations. Conference paper, Society of Petroleum Engineers (SPE) Symposium on Numerical Simulation of Reservoir Performance, Los Angeles, CA. https://doi.org/10.2118/5729-MS.

Vogel, T., & Císlerová, M. (1988). On the Reliability of Unsaturated Hydraulic Conductivity Calculated from the Moisture Retention Curve. *Transp. Porous Media*, 3(1), 1–15.

Vogel, T., van Genuchten, M.Th., & Císlerová, M. (2000). Effect of the Shape of the Soil Hydraulic Functions Near Saturation on Variably-Saturated Flow Predictions. *Adv. Water Resour.*, 24(2), 133–144. https://doi.org/10.1016/S0309-1708(00)00037-3.

Vrugt, J.A., Hopmans, J.W., & Šimůnek, J. (2001a). Calibration of a Two-Dimensional Root Water Uptake Model. *Soil Sci. Soc. Am. J.*, 65(4), 1027–1037.

Vrugt, J.A., van Wijk, M.T., Hopmans, J.W., & Šimůnek, J. (2001b). One-, Two-, and Three-Dimensional Root Water Uptake Functions for Transient Modeling. *Water Resour. Res.*, 37(10), 2457–2470.

Young, D.M., & Jea, K.C. (1980). Generalized Conjugate-Gradient Acceleration of Nonsymmetrizable Iterative Methods. *Linear Algebra Appl.*, 34, 159–194.

Tab. S1. Specifications of the Ocelote and Puma HPC clusters.

Ocelote HPC	
Model	Lenovo NeXtScale nx360 M5
Year	2016 (2018 P100 nodes)
Node Count	400
Total System Memory (TB)	82.6TB
Processors	2x Xeon E5-2695v3 14-core (Haswell) 2x Xeon E5-2695v4 14-core (Broadwell) 4x Xeon E7-4850v2 12-core (Ivy Bridge)
Cores/Node (schedulable)	28c (48c-high-memory node)
Total Cores	11528
Processor Speed	2.3GHz (2.4GHz - Broadwell CPUs)
Memory/Node	192GB (2TB-high-memory node)
Accelerators	46 NVIDIA P100 (16GB) 15 NVIDIA K80
/tmp	~840 GB spinning (/tmp is part of root filesystem)
HPL Rmax (TFlop/s)	382
OS	CentOS 6
Interconnect	FDR InfiniBand node-node; 10Gb Ethernet node-storage
Puma HPC	
Model	Penguin Altus XE2242
Year	2020
Node Count	236 CPU, 8 GPU, 2 High Memory
Total System Memory (TB)	128TB
Processors	2x AMD EPYC 7642 48-Core (Rome)
Cores/Node (schedulable)	94c
Total Cores	23616
Processor Speed	2.4GHz
Memory/Node	512GB (3TB - High-memory nodes)
Accelerators	29 NVIDIA V100S
/tmp	~1640 GB NVMe (/tmp is part of root filesystem)
OS	CentOS 7
Interconnect	1x 25Gb/s Ethernet RDMA (RoCEv2) 1x 25Gb/s Ethernet storage