Solving Richards equation: theoretical challenges and parallel implementation†

D. Bertaccini[1,2], P. D'Ambra[2], F. Durastante[3,2], S. Filippone[4,2]

[1] Department of Mathematics, University of Rome "Tor Vergata", [2] Istituto per le Applicazioni del Calcolo "M. Picone", National Research Council of Italy, [3] Department of Mathematics, University of Pisa, [4] Department of Civil and Computer Engineering, University of Rome "Tor Vergata"

fabio.durastante@unipi.it

†This work is supported by the Horizon 2020 Project Energy oriented Centre of Excellence: Toward Exascale for Energy (EoCoE-II), project ID 824158.

The Model Problem

Groundwater flow in the unsaturated zone is a **highly nonlinear phenomenon** that can be modeled by Richards equation:

$$\rho\,\phi\frac{\partial s(p)}{\partial t}-\nabla\cdot K(p)\nabla p-\frac{\partial K(p)}{\partial z}=f,$$

where $p(t)$ is the pressure head at time t, $s(p)$ is water saturation at pressure head p, ρ is water density, ϕ is porosity of the medium, $K(p)$ the hydraulic conductivity, f represents any water source terms and z is elevation.

Main applications: Create a realistic geothermal reservoir model for *energy production*, and management of storage reservoirs.

The solution procedure

We apply a quasi Newton method, as implemented in the **Kinsol library**, for the solution, at each time step, of the nonlinear system of equations obtained from a cell centered finite-difference discretizazion over a uniform mesh:

$$\begin{aligned}\boldsymbol{\Phi}(p^{(l)}_{i,j,k}) =&\frac{\rho\phi}{\Delta t}\left(s\left(p^{(l)}_{i,j,k}\right)-s\left(p^{(l-1)}_{i,j,k}\right)\right)+q^{(l)}_{i+1/2,j,k}-q^{(l)}_{i-1/2,j,k}\\&+q^{(l)}_{i,j+1/2,k}-q^{(l)}_{i,j-1/2,k}+q^{(l)}_{i,j,k+1/2}-q^{(l)}_{i,j,k-1/2}+f_{i,j,k}\equiv 0.\end{aligned}$$

for $i,j,k=1,\dots,\mathbf{N}-2$.

The Newton method works by computing increments $\mathbf{d}_r$ as the solution of:

$$J(\mathbf{p}_r)\mathbf{d}_r=-\boldsymbol{\Phi}(\mathbf{p}_r),$$

where $J(\mathbf{p}_r)$ is the Jacobian matrix of $\boldsymbol{\Phi}$, and $\mathbf{p}_r$ is the current iterate of pressure head.

Tasks:

- Finding **spectral information** on the Jacobian matrices;
- **Parallel** and **efficient** solution of the (right) preconditioned linear system

$$JM^{-1}(M\mathbf{d}_r)=-\boldsymbol{\Phi}(\mathbf{p}_r);$$

- **Minimize** the **number** of Jacobians and preconditioners **builds**;
- Enable *runs on thousand of computing cores*.

Spectral properties of the Jacobian sequence

To obtain the matrix M we investigate the **distribution of the eigenvalues**, $\{\lambda_j(J_\mathbf{N})\}_{j=1}^{\mathbf{N}}$, for the Jacobian matrix $J_\mathbf{N}$ of size $N=N_xN_yN_z$. We look for a measurable function $f:D\subset\mathbb{R}^k\to\mathbb{C}$ to associate to the sequence $\{J_\mathbf{N}\}_\mathbf{N}$ for which we can prove the following *asymptotic* relation

$$\lim_{\mathbb{N}\to\infty}\frac{1}{N}\sum_{i=1}^{N}F(\lambda_i(J_\mathbf{N}))=\frac{1}{\mu_k(D)}\int_D F(f(\mathbf{x}))\mathrm{d}\mathbf{x},\qquad\forall\,F\in C_c(\mathbb{C}),$$

where $\mu_k(\cdot)$ represent the Lebesgue measure on $\mathbb{R}^k$, $0<\mu_k(D)<\infty$, and $C_c(\mathbb{C})$ is the space of continuous functions with compact support.

The informal idea is that, if we assume that N is large enough, then the **eigenvalues** of the matrix $J_\mathbf{N}$, except possibly for $o(N)$ outliers, are **approximately equal** to the **samples of** f over a uniform grid in D.

Main result and construction of the auxiliary sequence

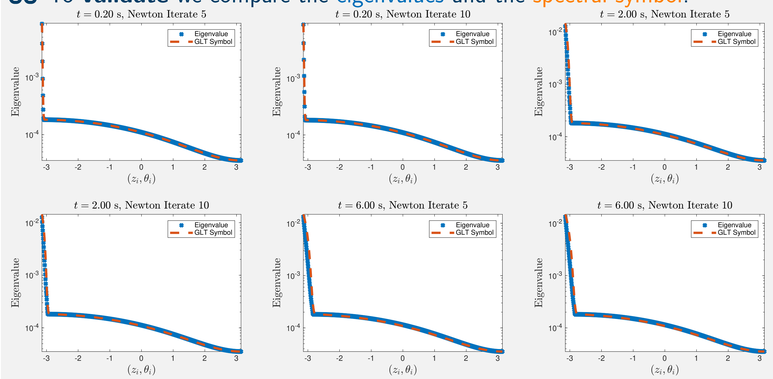
Theorem. The sequence $\{J_\mathbf{N}\}_\mathbf{N}$ obtained from the Richards equation discretization, for $K(p)$, $s(p)$ is **distributed in the sense of the eigenvalues** as the function

$$\begin{aligned}f(\mathbf{x},\theta)=C\rho\phi s'(\mathbf{p}^{(k,j)}(\psi(\mathbf{x})))\qquad&\mathbf{x}\in[0,1]^3,\ \theta\in[-\pi,\pi]^3,\\+K(\mathbf{p}^{(k,j)}(\psi(\mathbf{x})))(8-2\cos(\theta_1)-2\cos(\theta_2)-2\cos(\theta_3)),&\end{aligned}$$

where $\psi(\mathbf{x})$ is the function mapping the cube $[0,1]^3$ to the physical domain and $C=\lim_{\mathbf{N},N_T\to\infty}\frac{\mathbf{h}}{\Delta t}$. □

The *asymptotic* behavior is **dominated** by the **diffusive operator**.

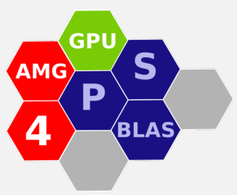
To **validate** we compare the eigenvalues and the spectral symbol:



The **key idea** is that we can use the discretization of the **diffusion operator** to **precondition** the system, actually an *Algebraic Multigrid preconditioner* (AMG) built on it!

PSCToolkit (Parallel Sparse Computation Toolkit)

To **build**, **precondition** and **solve** these systems we use a suite of massively parallel sparse solvers, preconditioners, and sparse linear algebra routines *we have developed* and that is called **PSCToolkit**. Of the whole toolkit we use the **PSBLAS** and **AMG4PSBLAS** libraries to handle data parallelism and iteratively solving the sequence of linear systems.



We use here for the first time a new set of interfaces we have written for the SUNDIALS/KINSOL library to employ their version of the modified Newton solvers making it rely on our implementations for linear algebra, linear solvers and preconditioners.

VSMATCH3 **V**-Cycle with 1 sweep of Hybrid Forward/Backward Gauss-Seidel (HFBGS) smoother, and **S**moothed aggregation based on **3** sweeps of graph **match**ing,

VSBM **V**-Cycle with 1 sweep of HFBGS smoother, and **S**moothed aggregation based on the scheme by **B**rezina, **M**andel and Vaněk,

with Block-Jacobi ILU(0) preconditioned Flexible Conjugate Gradient coarse solver.

Further information: `https://psctoolkit.github.io/`

P. D'Ambra, F. Durastante and S. Filippone, AMG Preconditioners for Linear Solvers towards Extreme Scale, SIAM J. Sci. Comput. **43** (2021), no. 5, S679–S703.

Test problem and employed machine

We solve the Richards equation with **upstream averages** on a parallelepipedal domain Ω of size $[0,L_x]\times[0,L_y]\times[0,L]$, we apply water at height $z=L$ such that

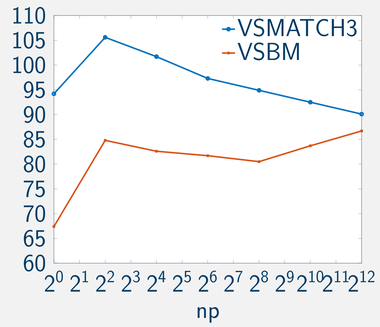
$$\begin{aligned}p(x,y,L,t)=\frac{1}{\alpha}\ln\Big[&\exp(\alpha h_r)\\&+(1-\exp(\alpha h_r))\chi_{[\frac{a}{4},\frac{3a}{4}]\times[\frac{b}{4},\frac{3b}{4}]}(x,y,z)\Big],\end{aligned}$$

where we denote by χ_Ω the characteristic function of the set Ω, and is fixed to the value. The initial condition is given by $p(x,y,z,0)=h_r$. In all cases we run the simulation for $t\in[0,2]$ and $N_t=10$.

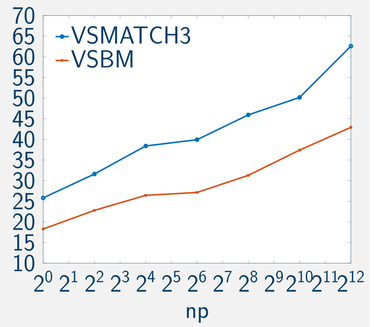
We use the **MARCONI 100** machine: the new accelerated cluster based on IBM Power9 architecture and Volta NVIDIA GPUs, ranked 14th in the June 2021 TOP500 list.

Weak scaling - $\sim 410M$ dofs on up to 4096 cores

We perform a weak scaling analysis by fixing the number of degree of freedoms for a processor to be $\mathbf{N}(k)=(2^kN_x,2^kN_y,N_z)$ for $np=4^k$ processors, $k=0,\dots,6$, for $N_x=N_y=50$, and $N_z=40$, for a domain $\Omega(k)=[0,2^k\times4.0]^2\times[0,1.0]$.



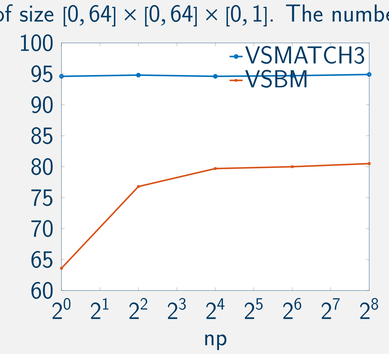
Average number of iterations



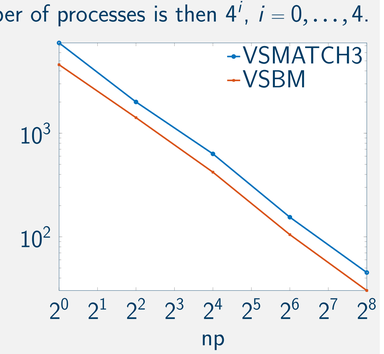
Total time to solution (s)

Strong scaling - $\sim 25M$ dofs from 1 to 256 cores

We fix the target size of the system to $800\times800\times40\approx2\cdot10^7$ dofs on a domain of size $[0,64]\times[0,64]\times[0,1]$. The number of processes is then 4^i, $i=0,\dots,4$.



Average number of iterations



Total time to solution (s)