Isogeometric alternating-directions CO_2 sequestration simulator

Askold Vilkha[1][0000-0001-9272-9082],

Marcin Łoś[1][0000-0002-8426-6345],
Maciej Paszyński[1][0000-0001-7766-6052]

[1]AGH University of Krakow, Poland
askoldvilkha@gmail.com, los@agh.edu.pl, paszynsk@agh.edu.pl

Abstract. This paper presents the implementation and preliminary results of a two-phase CO_2 sequestration simulator using the IGA-ADS (IsoGeometric Analysis using Alternating Directions Solver) framework. We formulate the problem based on Darcy's Law for two-phase flow in porous media, solving coupled equations for pressure and gas saturation. The spatial discretization employs B-spline basis functions with explicit time stepping for saturation and implicit solving for pressure using the MUMPS direct solver. We demonstrate the solver's capabilities on three reservoir configurations with heterogeneous porosity and permeability fields. Performance analysis on mesh sizes up to 256x256 points shows that pressure computation dominates the computational cost. The solver successfully handles physically realistic parameters and can efficiently utilize HPC resources via parallelization supported by GALOIS library.

Keywords: Isogeometric finite element method · CO_2 sequestration · Alternating direction solver · L2 projections · Non-stationary simulations

1 Introduction

Given the significant influence of carbon dioxide (CO_2) on global warming, the development of carbon capture strategies is of the highest importance. One of the prevalent strategies is the process called CO_2 sequestration. The main idea of this process is to directly capture the CO_2 from the emission sources, such as power plants and factories, and store it in the underground porous structures of both natural and human-made origin (abandoned oil wells, mines, aquifers) [11,9,16]. This strategy is more beneficial than others (ocean storage, mineralization, direct

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29912-3_7

air capture) and it is expected to be used until more environmentally-friendly alternatives for energy production will come to wide usage [12,5].

We present the implementation and preliminary testing of a simplified two-phase flow CO_2 sequestration model within the IGA-ADS (IsoGeometric Analysis using Alternating Directions Solver) framework [22]. This framework is a Finite-Element Method (FEM) solver, which has been successfully applied to problems like heat transfer [22], Maxwell's equations solver [21], and tumor growth simulations [18]. In the context of the CO_2 sequestration problem, four common situations are outlined: structural, dissolution, capillar, and mineralization [12]. Our solver can be applied to the first two of them, as it does not take into account the chemical reactions in the reservoir, but can effectively simulate the physical behavior of the pumped gas in the porous structure and in the case of pumping into the liquid brine solution. We aim to provide a fast and reliable solver for the simplified benchmark problem (similar to [7,17]) that can be used as a building block for more complex models in the future. The main contributions of this paper are: a) a clear formulation and implementation of the coupled pressure–saturation system within the IGA-ADS C++ framework; b) a performance analysis identifying the pressure solve as the dominant cost for realistic mesh sizes; c) preliminary simulation results on heterogeneous porosity and permeability fields confirming physically plausible behavior.

2 CO_2 sequestration problem formulation

Following the previous works [16,7,17], we use the analytical expression for the CO_2 sequestration problem (Darcy's Law).

$$\phi \partial_t S_w = \nabla \cdot \left(\frac{S_w}{\mu_w} K \left(\nabla p - \rho_w \boldsymbol{g} \right) \right) + q_w, \tag{1}$$

$$\phi \partial_t S_g = \nabla \cdot \left(\frac{S_g}{\mu_g} K \left(\nabla p - \rho_g \boldsymbol{g} \right) \right) + q_g, \tag{2}$$

where S_w and S_g are the brine solution and gas saturations, respectively, μ_w and μ_g are the viscosities of the brine solution and gas, p is the pressure, ρ_w and ρ_g are the densities of the brine solution and gas, g is the gravitational acceleration, and q_w and q_g are the source terms, which represent the pumping locations and intensities for the brine solution and gas, respectively. The porosity ϕ is the fraction of the total volume occupied by the pores in the reservoir, and

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29912-3_7

the permeability K describes the ability of the porous medium to allow fluids to flow through it [13]. *Note:* Our work uses the same set of assumptions as the benchmark problem presented in [7], i.e. isothermal conditions, negligible capillary pressure, constant fluid properties, and linear relative permeability. A trivial constraint that $S_w + S_g = 1$ is also applied. Taking this into account, we can rewrite the equation (1) as follows:

$$-\phi\partial_t S_g = \nabla \cdot \left(\frac{1 - S_g}{\mu_w} K \left(\nabla p - \rho_w \boldsymbol{g} \right) \right) + q_w. \tag{3}$$

Hence, we have a system of two equations (3) and (2) with two unknowns S_g and p, and by summing up these equations we obtain an equation for pressure:

$$\nabla \cdot \left(\left(\frac{1 - S_g}{\mu_w} + \frac{S_g}{\mu_g} \right) K \nabla p \right) = \nabla \cdot \left(\boldsymbol{g} K \left(\frac{1 - S_g}{\mu_w} \rho_w + \frac{S_g}{\mu_g} \rho_g \right) \right) - (q_w + q_g). \tag{4}$$

We will then use this equation to compute pressure p after setting some initial saturation S_g^0. In each following time step, we compute pressure p^n based on the saturation S_g^n as follows:

$$\nabla \cdot \left(\left(\frac{1 - S_g^n}{\mu_w} + \frac{S_g^n}{\mu_g} \right) K \nabla p^n \right) = \nabla \cdot \left(\boldsymbol{g} K \left(\frac{1 - S_g^n}{\mu_w} \rho_w + \frac{S_g^n}{\mu_g} \rho_g \right) \right) - (q_w^n + q_g^n) \tag{5}$$

After computing p^n and S_g^n, we can compute S_g^{n+1} based on the pressure p^n as follows (using the forward Euler method):

$$\phi \frac{S_g^{n+1} - S_g^n}{\tau} = \nabla \cdot \left(\frac{S_g^n}{\mu_g} K \left(\nabla p^n - \rho_g \boldsymbol{g} \right) \right) + q_g^n, \tag{6}$$

which implies

$$S_g^{n+1} = S_g^n + \phi^{-1} \tau \nabla \cdot \left(\frac{S_g^n}{\mu_g} K \left(\nabla p^n - \rho_g \boldsymbol{g} \right) \right) + \phi^{-1} \tau q_g^n. \tag{7}$$

3 Explicit dynamics with isogeometric alternating direction solver

The equation (6) can be rewritten in the general form

$$\partial_t S - \mathcal{L}(S) = f, \tag{8}$$

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29912-3_7

where

$$\mathcal{L}(S) = \nabla \cdot \left(\frac{S}{\mu_g \phi} K \left(\nabla p - \rho_g \boldsymbol{g} \right) \right), \tag{9}$$

and

$$f = \phi^{-1} q_g. \tag{10}$$

For a non-stationary problem of the form (8) with some initial state S_0 and boundary conditions, where $\mathcal{L}$ is well-posed partial differential operator, we employ B-spline basis functions of order r and continuity C^{r-1} for spatial discretization

$$S(x, y) \approx \sum_{i,j} S_{i,j} B^x_{i,r}(x) B^y_{j,r}(y). \tag{11}$$

For the time discretization, we employ the explicit dynamics method

$$S_{n+1} = S_n + \tau \mathcal{L} S_n + \tau f_n. \tag{12}$$

We derive a variational formulation for computing L^2 products with test functions

$$(S_{n+1}, v)_{L^2} = (S_n + \tau \mathcal{L} S_n + \tau f_n, v)_{L^2}, \ \forall v = B^x_{k,r}(x) B^y_{l,r}(y), \tag{13}$$

Employing B-spline basis functions for testing implies a sequence of isogeometric L^2 projections to be computed in every time step.

$$S_n \approx \sum_{i,j} S^{i,j}_n B^x_{i,r}(x) B^y_{j,r}(y). \tag{14}$$

The system of linear equations to solve in every time step looks like

$$\begin{aligned} &\sum_{i,j} S^{i,j}_{n+1} (B^x_{i,r}(x) B^y_{j,r}(y), B^x_{k,r}(x) B^y_{l,r}(y))_{L^2} = \\ &(\sum_{i,j} S^{i,j}_n B^x_{i,r}(x) B^y_{j,r}(y) + \tau \sum_{i,j} S^{i,j}_n \mathcal{L}(B^x_{i,r}(x) B^y_{j,r}(y)) + f_n, B^x_{k,r}(x) B^y_{l,r}(y)) \end{aligned} \tag{15}$$

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29912-3_7

For all k, l. Hence, we compute a sequence of isogeometric L^2 projections in every time step. It implies the following system of linear equations

$$\begin{bmatrix} (B^x_{1,r}B^y_{1,r}, B^x_{1,r}B^y_{1,r}) & \cdots & (B^x_{1,r}B^y_{1,r}, B^x_{N_x,r}B^y_{N_y,r}) \\ \vdots & \ddots & \vdots \\ (B^x_{N_x,r}B^y_{N_y,r}, B^x_{1,r}B^y_{1,r}) & \cdots & (B^x_{N_x,r}B^y_{N_y,r}, B^x_{N_x,r}B^y_{N_y,r}) \end{bmatrix} \begin{bmatrix} S^{1,1}_{n+1} \\ \vdots \\ S^{N_x,N_y}_{n+1} \end{bmatrix} = \begin{bmatrix} b^{1,1}_n \\ \vdots \\ b^{N_x,N_y}_n \end{bmatrix} \tag{16}$$

where $b^{k,l}_n$ corresponds to the right-hand side of the equation (15).

The Gram matrix build with B-splines over regular 2D domain $\Omega = \Omega_x \times \Omega_y$ has a Kronecker product structure

$$\begin{aligned} \mathcal{M}_{ijkl} &= (B_{ij}, B_{kl})_{L^2} = \int_\Omega B_{ij} B_{kl} \,\mathrm{d}\Omega = \int_\Omega B^x_i(x) B^y_j(y) B^x_k(x) B^y_l(y) \,\mathrm{d}\Omega \\ &= \int_\Omega (B_i B_k)(x)\,(B_j B_l)(y) \,\mathrm{d}\Omega = \left(\int_{\Omega_x} B_i B_k \,\mathrm{d}x \right) \left(\int_{\Omega_y} B_j B_l \,\mathrm{d}y \right) = \mathcal{M}^x_{ik} \mathcal{M}^y_{jl} \end{aligned} \tag{17}$$

In other words $\mathcal{M} = \mathcal{M}^x \otimes \mathcal{M}^y$ and the inverse $\mathcal{M}^{-1} = (\mathcal{M}^x)^{-1} \otimes (\mathcal{M}^y)^{-1}$. In our case, $\mathcal{M}^x$ and $\mathcal{M}^y$ are pentadiagonal (two non-zero diagonals on each side of the main diagonal) due to local B-splines support with $r = 2$, so the system can be solved using a banded LU-factorization (a logical extension of the Thomas algorithm) in $O(N)$ time, where N is the total number of elements in the mesh.

4 Direct solver for pressure solution

To simplify notation, we introduce the *total mobility* λ and the *effective gravity coefficient* ρ^*, both depending on the current saturation S^n_g:

$$\lambda(S^n_g) = \frac{1 - S^n_g}{\mu_w} + \frac{S^n_g}{\mu_g}, \qquad \rho^*(S^n_g) = \frac{1 - S^n_g}{\mu_w}\rho_w + \frac{S^n_g}{\mu_g}\rho_g, \tag{18}$$

so that the pressure equation (5) takes the compact form

$$\nabla \cdot \big(\lambda(S^n_g)\, K\, \nabla p^n\big) = \nabla \cdot \big(\boldsymbol{g}\, K\, \rho^*(S^n_g)\big) - (q^n_w + q^n_g). \tag{19}$$

The weak formulation of (19) seeks $p^n \in V_h$ such that

$$a^n(p^n,\, v) = \ell^n(v) \quad \forall\, v \in V_h, \tag{20}$$

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29912-3_7

where $(u,v)_{L^2} = \int_\Omega u(x,y)v(x,y)\,dxdy$ and the bilinear and linear forms are defined as

$$a^n(u,\,v) = \left(\lambda(S_g^n)\,K\,\nabla u,\;\nabla v\right)_{L^2}, \tag{21}$$

$$\ell^n(v) = \left(\boldsymbol{g}\,K\,\rho^*(S_g^n),\;\nabla v\right)_{L^2} + \left(q_w^n + q_g^n,\;v\right)_{L^2}. \tag{22}$$

The saturation and pressure fields are both approximated using B-spline basis functions:

$$S_g^n(x,y) = \sum_{i,j} S_n^{i,j}\,B_{i,r}^x(x)\,B_{j,r}^y(y), \qquad p^n(x,y) = \sum_{i,j} p_n^{i,j}\,B_{i,r}^x(x)\,B_{j,r}^y(y). \tag{23}$$

The permeability map $K(x,y)$ and porosity $\phi(x,y)$ are read from the input files (the same as configuration P10 from [17]) and interpolated linearly to the quadrature points. The source functions q_g^n and q_w^n represent pumping locations and intensities for gas (g) and brine (w), respectively.

Testing with $v = B_{k,r}^x(x)\,B_{l,r}^y(y)$ yields the linear system $\mathcal{S}\,\mathcal{P} = \mathcal{F}$, whose entries are

$$\mathcal{S}_{(i,j;\,k,l)} = a^n\left(B_{i,r}^x B_{j,r}^y,\;B_{k,r}^x B_{l,r}^y\right), \tag{24}$$

$$\mathcal{F}_{(k,l)} = \ell^n\left(B_{k,r}^x B_{l,r}^y\right), \tag{25}$$

and $\mathcal{P} = \left[p_n^{1,1},\;\ldots,\;p_n^{N_x,N_y}\right]^\top$. Unlike the mass matrix in the saturation solver, the stiffness matrix $\mathcal{S}$ does not have a Kronecker product structure due to the spatially varying coefficients $\lambda(S_g^n)$ and K. Hence, we call the implicit solver for pressure, after assembling $\mathcal{S}$ and $\mathcal{F}$, at each time step for the current saturation S_g^n (this step is called "assemble problem"). Zero Dirichlet boundary conditions are enforced by zeroing the corresponding rows, placing 1 on the diagonal, and setting the right-hand side entry to 0. The resulting system is solved with MUMPS [1,2].

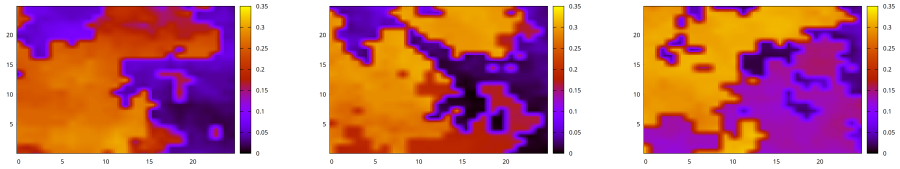
5 Results

5.1 Simulation Setup

We present results for three different test configurations (K1, K2, K3) with varying porosity and permeability maps. The porosity and permeability distributions for each configuration are illustrated in Figures 1 and 2, respectively. For all our

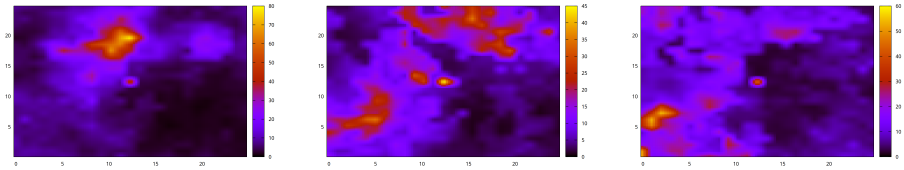
ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29912-3_7

simulations, if we use a permeability map, we use the corresponding porosity map. The computational domain is a 2D rectangular region of size $50m \times 50m$. The physical parameters used in the simulations are summarized in Table 1. The simulations are performed using a uniform mesh with appropriate boundary conditions: no-flux conditions, and pressure boundary conditions on the left, right, top, and bottom boundaries. The CO_2 injection is modeled through a source term located at the center of the domain, representing an injection well.



(a) Configuration K1 (b) Configuration K2 (c) Configuration K3

Fig. 1: Porosity distributions for three different reservoir configurations. The values range from 0 to 1.



(a) Configuration K1 (b) Configuration K2 (c) Configuration K3

Fig. 2: Permeability distributions for three different reservoir configurations. The units are in mDarcy, in simulation we use SI units (1 mDarcy $\simeq 10^{-15}$ m^2).

5.2 IGA-ADS Direct Solver

For this paper, we completed simulations for the configurations K1, K2, and K3 both for porosity-only and porosity-permeability cases. However, because of the limited space and to avoid copyright issues[1], we only present the results for the

[1] Results for the porosity-permeability case will be shown at the conference presentation and in the paper that has been recently submitted to Computers & Mathematics with Applications under the title: "CO2 sequestration hybrid solver using isogeomet-

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29912-3_7

	SI
ρ_g	479 (kg/m^3)
ρ_w	1045 (kg/m^3)
g	9.81 (m/s^2)
μ_g	3.95×10^{-5} (Pa $\cdot$ s)
μ_w	25.35×10^{-5} (Pa $\cdot$ s)
K_A	1×10^{-15} (m^2)

Table 1: Simulation parameters in SI units

porosity-only case in this paper. Simulations with heterogeneous permeability show similar gas flow structure but with a strong preference for high permeability regions, which is consistent with physical expectations. The simulation parameters are set as follows: time step size $\tau = 5000$ seconds, number of time steps = 500, and source term defined as a circle of radius 3 m at the center of the domain with a strength of $q_g = 1 \times 10^{-6}$ (dimensionless units of S_g) per iteration. For this configuration, the estimated maximum time step size for stability is around 7500 seconds. Larger time steps are possible but will reduce the accuracy. We have observed an inverse linear rule regarding the maximum time step size for stability w. r. t. the grid size. We also advise using smaller time step sizes for larger values of permeability as it increases the gas flow velocity. The results of the simulations are presented in Figures 3, 4, 5.

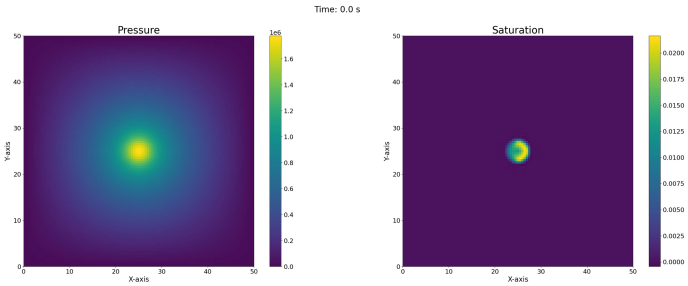
5.3 Timing and Performance Analysis

We started with a cost decomposition test to identify the most computationally expensive parts of the solver. The simulations were run for 1000 iterations each with the same parameters as in the previous section, but with varying grid sizes on a single node of the ARES supercomputer in CYFRONET (using 1 CPU core, CPU Model: Intel Xeon Platinum 8268 CPU @ 2.90 GHz). These measurements are summarized in Figure 6. As expected, the cost is dominated by the pressure part, which requires calling the MUMPS solver at every time step of the simulation to solve the stationary pressure equation.

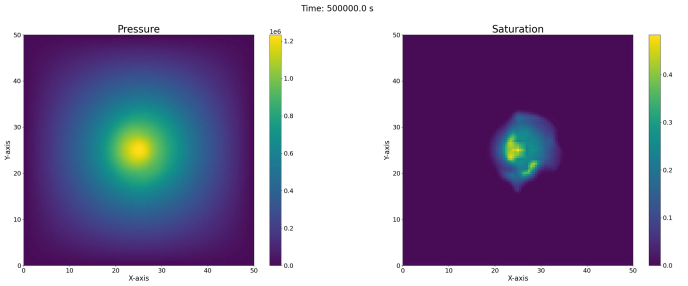
Secondly, we have conducted a strong scaling test to evaluate the parallel performance of the solver (Fig. 7). The parallelization is done using the shared memory approach from GALOIS library [10] and is applied to saturation and

ric alternating-directions and collocation-based robust variational physics informed neural networks (IGA-ADS-CRVPINN)"

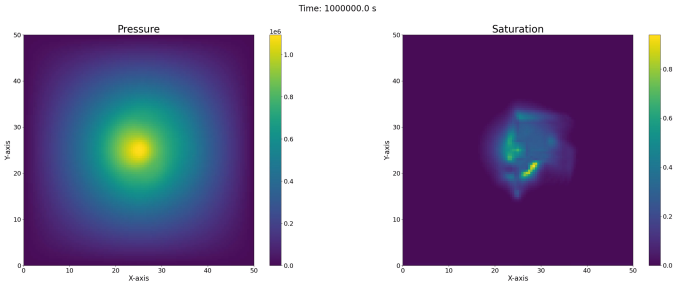
ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29912-3_7



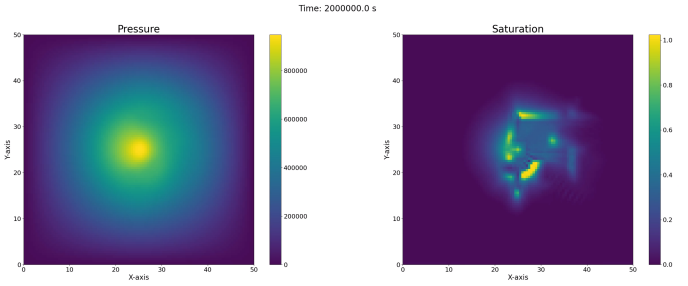
(a) t = 0 s



(b) t = 500,000 s (= 100 time steps)



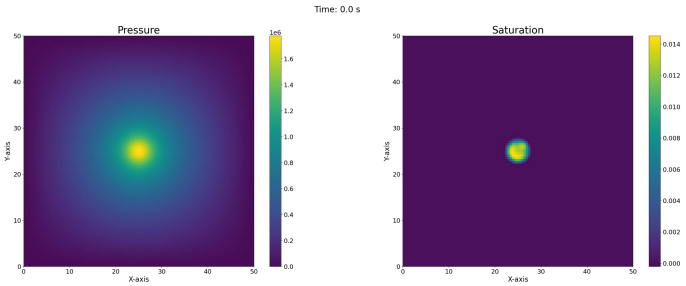
(c) t = 1,000,000 s (= 200 time steps)



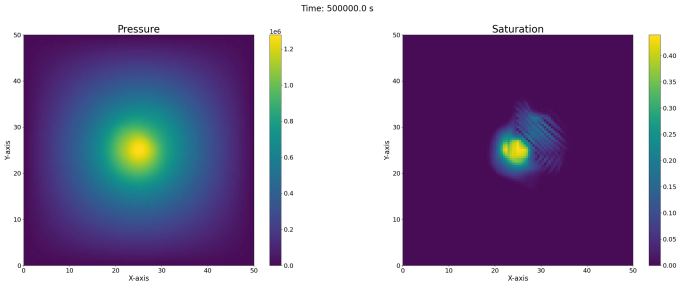
(d) t = 2,000,000 s (= 400 time steps)

Fig. 3: CO_2 saturation evolution for configuration K1 with uniform permeability at different time steps.

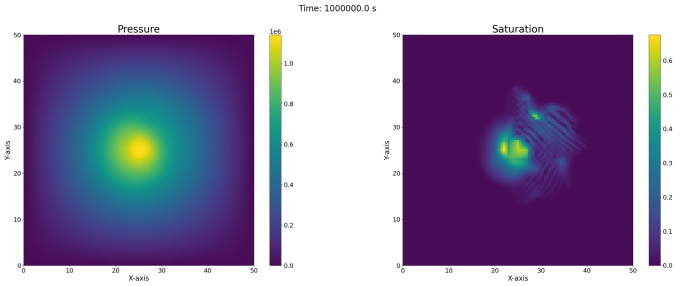
ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29912-3_7



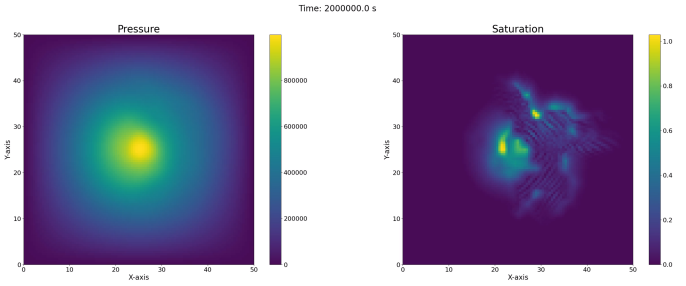
(a) t = 0 s



(b) t = 500,000 s (= 100 time steps)



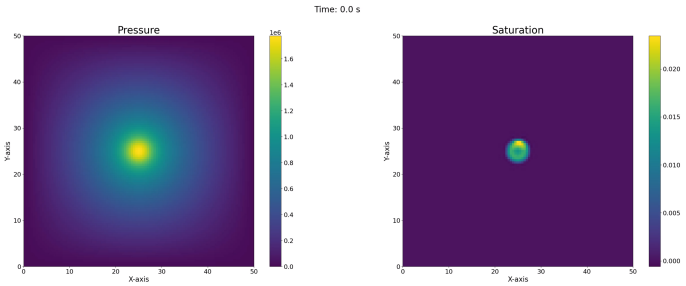
(c) t = 1,000,000 s (= 200 time steps)



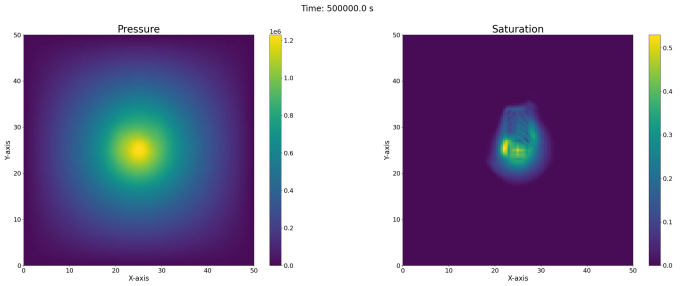
(d) t = 2,000,000 s (= 400 time steps)

Fig. 4: CO_2 saturation evolution for configuration K2 with uniform permeability at different time steps.

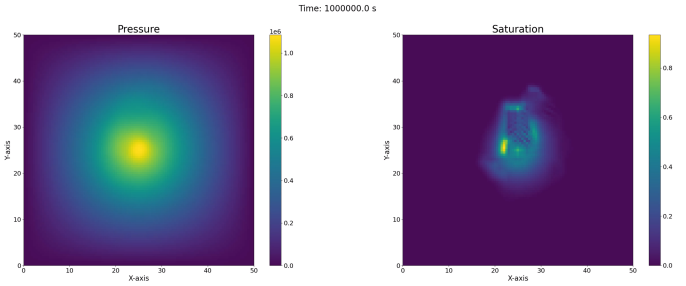
ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29912-3_7



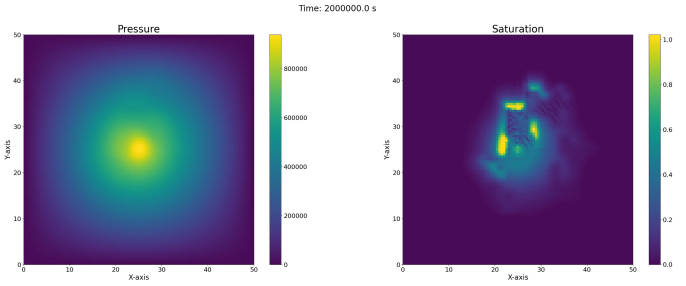
(a) t = 0 s



(b) t = 500,000 s (= 100 time steps)



(c) t = 1,000,000 s (= 200 time steps)



(d) t = 2,000,000 s (= 400 time steps)

Fig. 5: CO_2 saturation evolution for configuration K3 with uniform permeability at different time steps.

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29912-3_7

pressure integration. The saturation solver, initialization, and output are single-threaded since they are significantly cheaper than other parts of the solver. MUMPS solver is also single-threaded since it uses a different parallelization technique. The speedup for the saturation integration is almost ideal, while the pressure integration shows much worse speedup (4-5 times smaller than the saturation). The reason for this is the process called "assemble problem", which builds the linear system for the MUMPS solver and is parallelized over boundary degrees of freedom (400 for the grid size of 100×100).

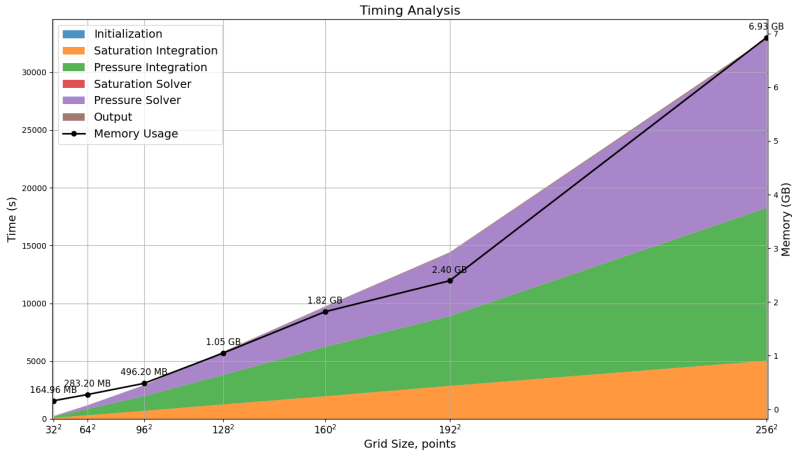


Fig. 6: Timings of IGA-ADS solver for various mesh sizes (regular grids). (*Note:* Initialization, Saturation solver, and Output are not visible due to their small contributions.) As one can see, the most computationally expensive part are pressure integration and solution, which contribute to about 80% of the total time (for 128×128 grid: $4494/5809 = 77.4$ %). The plot also shows the RAM memory usage for each grid size (ranging from 0.16 GB to 6.93 GB).

6 Conclusions and Future Work

In this work, we have demonstrated some preliminary results of our CO_2 sequestration problem solver within the IGA-ADS framework with the primary focus on the computational methodology rather than the ideal geophysical model. It has shown the capability to solve varying cases of the simpified direct problem

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29912-3_7

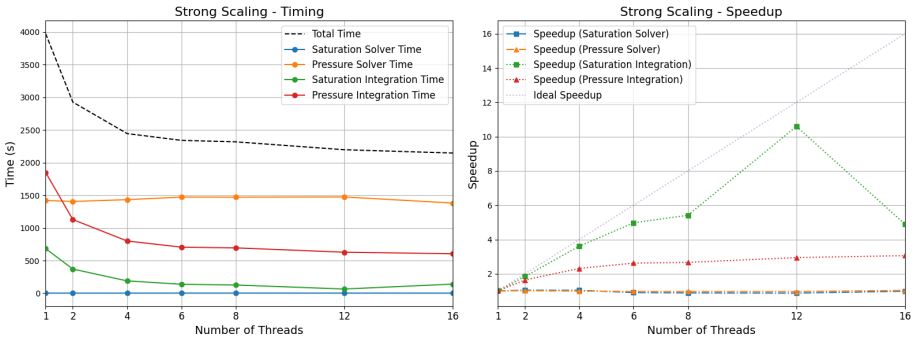


Fig. 7: Strong scaling of the solver for a fixed grid size of 100×100 and 1000 iterations. The MUMPS solver was run only on 1 CPU core, while all the other parts of the solver were parallelized with GALOIS using 1 to 16 CPU cores.

and provides a solid foundation for implementation of more complex features in the future. The solver scales from very limited computational resources (AMD Ryzen 5 5500 U CPU and 10 GB of RAM) to usage of a single node of the ARES supercomputer. Parallelization of the CPU workload is supported by the GALOIS library [10]. The sample results correspond to the expected behavior of the system, and the solver works well with the parameter values provided in the literature [7,17]. Saturation values remain within $[0, 1]$ range, and both saturation and pressure match the patterns from [17]. The presented simulations cover $2,5 \times 10^6$ seconds, ($\approx$ 29 days). Additional runs with the same parameter set were conducted for longer timescales (up to 5×10^6 seconds, ≈ 58 days) without divergence, producing plume patterns qualitatively similar to those reported in [7] for comparable timescales. The gas saturation flow mirrors the porosity map, which is expected since more porous regions should be able to accommodate more gas and thus have higher saturation values.

The future work will involve solving the inverse problem—identifying the optimal location for the CO_2 injection well and the adjustable parameter values for a given reservoir. We are now developing a hybrid solver based on Collocation-based Robust Variational Physics Informed Neural Networks model from a recent paper by M. Łoś et. al. [20] to solve the pressure while still relying on the IGA-ADS solver for saturation. We may also develop a parallel distributed memory version of the solver with the library described in [19], consider including *hp*-adaptive FEM solvers [14,15,6], or applying inverse solvers [4,8,3] for a better fit of the model parameters.

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29912-3_7

7 Acknowledgements

The work of Askold Vilkha and Maciej Paszyński has received funding from the European Union's Horizon Europe research and innovation programme under the Marie Sklodowska-Curie grant agreement No 101119556. We gratefully acknowledge Polish high-performance computing infrastructure PLGrid (HPC Center: ACK Cyfronet AGH) for providing computer facilities and support within computational grants no. PLG/2024/017915 and no. PLG/2026/019371.

References

1. Amestoy, P., Buttari, A., L'Excellent, J.Y., Mary, T.: Performance and Scalability of the Block Low-Rank Multifrontal Factorization on Multicore Architectures. ACM Transactions on Mathematical Software **45**, 2:1–2:26 (2019)
2. Amestoy, P., Duff, I.S., Koster, J., L'Excellent, J.Y.: A fully asynchronous multifrontal solver using distributed dynamic scheduling. SIAM Journal on Matrix Analysis and Applications **23**(1), 15–41 (2001)
3. Barabasz, B., Gajda-Zagórska, E., Migórski, S., Paszyński, M., Schaefer, R., Smołka, M.: A hybrid algorithm for solving inverse problems in elasticity. Int. J. Appl. Math. Comput. Sci. **24**(4), 865–886 (Dec 2014)
4. Barabasz, B., Migórski, S., Schaefer, R., Paszyński, M.: Multi-deme, twin adaptive strategy hp-hgs. Inverse Problems in Science and Engineering **19**(1), 3–16 (2011)
5. Bielinski, A.: Numerical simulation of CO 2 sequestration in geological formations. No. 155 in Mitteilungen / Institut für Wasserbau, Universität Stuttgart, Inst. f. Wasserbau, Stuttgart (2007)
6. Demkowicz, L., Gatto, P., Kurtz, J., Paszyński, M., Rachowicz, W., Bleszyński, E., Bleszyński, M., Hamilton, M., Champlin, C., Pardo, D.: Modeling of bone conduction of sound in the human head using hp-finite elements: Code design and verification. Computer Methods in Applied Mechanics and Engineering **200**(21), 1757–1773 (2011)
7. Ebigbo, A., Class, H., Helmig, R.: CO2 leakage through an abandoned well: problem-oriented benchmarks. Computational Geosciences **11**(2), 103–115 (Jun 2007)
8. Gajda-Zagórska, E., Schaefer, R., Smołka, M., Paszyński, M., Pardo, D.: A hybrid method for inversion of 3d dc resistivity logging measurements. Natural computing **14**(3), 355–374
9. Holloway, S.: STORAGE OF FOSSIL FUEL-DERIVED CARBON DIOXIDE BENEATH THE SURFACE OF THE EARTH. Annual Review of Environment and Resources **26**(Volume 26, 2001), 145–166 (Nov 2001)

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29912-3_7

10. Jatala, V., Dathathri, R., Gill, G., Hoang, L., Nandivada, V.K., Pingali, K.: A study of graph analytics for massive datasets on distributed multi-gpus. In: 2020 IEEE International Parallel and Distributed Processing Symposium (IPDPS). pp. 84–94. IEEE (2020)
11. Lackner, K.S.: A Guide to CO2 Sequestration. Science **300**(5626), 1677–1678 (Jun 2003), publisher: American Association for the Advancement of Science
12. Massarweh, O., Abushaikha, A.S.: CO2 sequestration in subsurface geological formations: A review of trapping mechanisms and monitoring techniques. Earth-Science Reviews **253**, 104793 (Jun 2024)
13. Nishiyama, N., Yokoyama, T.: Permeability of porous media: Role of the critical pore size. Journal of Geophysical Research: Solid Earth **122**(9), 6955–6971 (2017)
14. Pardo, D., Demkowicz, L., Torres-Verdín, C., Paszynski, M.: A self-adaptive goal-oriented hp-finite element method with electromagnetic applications. part ii: Electrodynamics. Computer Methods in Applied Mechanics and Engineering **196**(37), 3585–3597 (2007)
15. Paszyński, M.: On the parallelization of self-adaptive hp-finite element methods part ii. partitioning communication agglomeration mapping (PCAM) analysis. Fundamenta Informaticae **93**(4), 435–457 (2009)
16. Pruess, K., García, J.: Multiphase flow dynamics during CO2 disposal into saline aquifers. Environmental Geology **42**(2), 282–295 (Jun 2002)
17. Shokouhi, P., Kumar, V., Prathipati, S., Hosseini, S.A., Giles, C.L., Kifer, D.: Physics-informed deep learning for prediction of *CO*2 storage site response. Journal of Contaminant Hydrology **241**, 103835 (Aug 2021)
18. Siwik, L., Łoś, M., Kłusek, A., Dzwinel, W., Paszyński, M.: Tuning Two-Dimensional Tumor Growth Simulations. In: Proceedings of the 50th Computer Simulation Conference. Society for Modeling and Simulation International (SCS), University of Bordeaux, Bordeaux, France (2018)
19. Woźniak, M., Łoś, M., Paszyński, M., Dalcin, L., Calo, V.M.: Parallel fast isogeometric solvers for explicit dynamics. Computing and Informatics **36**(2), 423–448 (Jun 2017)
20. Łoś, M., Służalec, T., Maczuga, P., Vilkha, A., Uriarte, C., Paszyński, M.: Collocation-based robust variational physics-informed neural networks (crvpinns). Computers & Structures **316**, 107839 (2025)
21. Łoś, M., Woźniak, M., Pingali, K., Castillo, L.E.G., Alvarez-Aramberri, J., Pardo, D., Paszyński, M.: Fast parallel IGA-ADS solver for time-dependent Maxwell's equations. Computers & Mathematics with Applications **151**, 36–49 (Dec 2023)
22. Łoś, M.M., Woźniak, M., Paszyński, M., Lenharth, A., Hassaan, M.A., Pingali, K.: IGA-ADS: Isogeometric analysis FEM using ADS solver. Computer Physics Communications **217**, 99–116 (Aug 2017)

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29912-3_7