

Application of a Hierarchical Tensor Decomposition to the Dynamics of a Post-Pumping Saltwater Intrusion Experiment

Benedikt Schröter[†][0009-0005-7358-7978],
Julian Hilbert[†],
Arne Nägel[0000-0003-0015-4379]

1 Introduction

In densely populated coastal areas, an adequate supply with fresh water is highly important for agriculture, economics and society. A sound understanding of saltwater intrusion phenomena, in particular, if they are caused by artificial freshwater extraction from the subsurface, is a key factor for sustainable water management; see, for example [7] and references cited therein.

The motivation for this work is the lab-scale experiment conducted by [11]. A central challenge is to accurately predict the length of the saltwater wedge that develops within a freshwater aquifer during pumping. The authors also developed a numerical model that qualitatively describe the observed experimental behaviour and studied the model parameters. In the present study, this analysis is further extended. Our results are obtained by combining two domain decomposition algorithms: First, the LIMEX-multigrid solver [8] is applied to solve a transient problem in partially saturated medium. Second, a Hierarchical Tucker Decomposition [4] is used to construct a reduced order model. This combination allows an in-depth analysis of the parameters, and yields an improved agreement with the experimental data.

2 Methods

Governing equations and numerical methods. Simulations were carried out using the software *d³f++* (mnemonic for *distributed density driven flow*) [2], a finite-volume framework for modeling variably saturated, variable-density groundwater

Benedikt Schröter, Julian Hilbert, Arne Nägel
Modular Supercomputing and Quantum Computing, Goethe University Frankfurt, e-mail: {b.schroeter, j.hilbert, naegel}@em.uni-frankfurt.de

[†]These authors contributed equally to this work.

flow and solute transport, based on UG4 [9, 12]. Here, the effective saturation is expressed by the van Genuchten relationship [3]

$$S(\psi) = \left(\frac{1}{1 + \alpha|\psi|} \right)^{1 - \frac{1}{n}}, \psi = -p,$$

where α [Pa⁻¹] and n [-] are empirical parameters and p [Pa] denotes the pressure head. The relative permeability is given by

$$k_r(S) = \sqrt{S} \left(1 - (1 - S)^{1 - \frac{1}{n}} \right)^{2(1 - \frac{1}{n})}.$$

The Darcy flux is modified by the relative permeability to account for reduced flow in the unsaturated zone

$$\mathbf{v}_f = -k_r \frac{K}{\mu} (\rho \mathbf{g} - \nabla p),$$

where K [m²] is the intrinsic permeability, μ [Pa·s] the dynamic viscosity, ρ [kg·m⁻³] the fluid density, and \mathbf{g} [kg·m·s⁻²] the gravitational acceleration. The transient flow- and solute transport equations are then described, cf. [1], by

$$\begin{aligned} \frac{\partial \phi S \rho}{\partial t} + \nabla \cdot (\rho \mathbf{v}_f) &= Q_p \\ \frac{\partial \phi \rho S c}{\partial t} + \nabla \cdot (\rho \mathbf{v}_f c) - \nabla \cdot \left[\left(\phi S \rho D_m \mathbf{I} + (\alpha_l - \alpha_t) \frac{\mathbf{v}_f \mathbf{v}_f^T}{|\mathbf{v}_f|} + \alpha_t |\mathbf{v}_f| \mathbf{I} \right) \nabla c \right] &= Q_c, \end{aligned}$$

where ϕ [-] is the porosity, c [-] the salt concentration, D_m [m²·s⁻¹] the effective molecular diffusion coefficient, and α_l and α_t denote the longitudinal and transverse dispersivities [10], respectively. The right-hand side Q represents source and sink terms. The governing equations are discretized using a vertex-centered finite volume method. A dual mesh of control volumes is constructed from the primary grid, which is further subdivided into sub-control volumes (SCVs) within each element. Fluxes across sub-control volume faces (SCVFs) are computed using numerical approximations evaluated elementwise. Material and transport parameters are interpolated to SCVF centers, while the relative permeability k_r is evaluated at upwind nodes with respect to the Darcy flux \mathbf{v}_f to ensure numerical stability. Time integration is performed using the LIMEX scheme [8], an adaptive, linearly implicit extrapolation method. LIMEX combines variable step-size and order control, dependent on estimated truncation errors.

Post-pumping seawater intrusion experiment. We consider the post-pumping seawater intrusion experiment conducted by Stoeckl et al. [11]. This experiment reproduces the phenomenon of seawater advancing inland beyond pump location after pumping has ceased.

The physical model consists of a transparent acrylic tank with dimensions of 2.0 m × 0.2 m × 0.05 m, representing an unconfined coastal aquifer filled with quartz sand. A

constant water level was maintained on the left boundary to represent the sea, while a constant freshwater inflow on the right boundary mimicked the inland recharge zone. Starting from an equilibrium, pumping was applied at a constant rate of $1\text{m}^3\cdot\text{day}^{-1}$ for a duration of 135s. During and after the pumping process, the transient evolution of the saltwater intrusion was monitored visually. The intrusion toe length, defined as the horizontal position of the 0.5 isoline of salt concentration along the bottom domain boundary, was recorded as a function of time and used as the primary target variable for model calibration. During model calibration, the intrinsic permeability K , porosity ϕ , and van Genuchten parameters α and n were considered.

To reproduce the laboratory setup numerically, we construct a two-dimensional cross-sectional model of the experiment (see Fig. 1). The computational domain represents the longitudinal section of the aquifer, assuming uniform conditions in the traverse direction. Salt concentration is normalized between $c = 0.0$ (freshwater) and $c = 1.0$ (seawater), with densities of $996.9\text{kg}\cdot\text{m}^{-3}$ and $1020.9\text{kg}\cdot\text{m}^{-3}$, respectively.

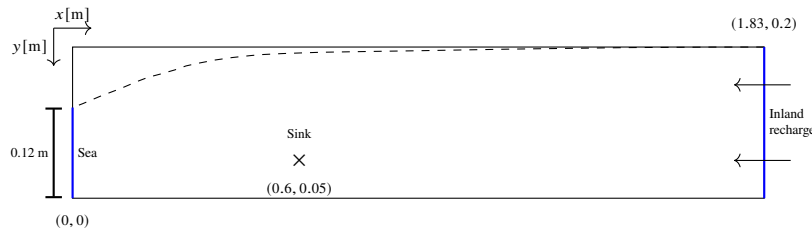


Fig. 1 Illustration of the computational domain. The dashed line indicates the water level at the start of pumping.

Hierarchical Tucker Representation. High-dimensional parameter studies often suffer from the curse of dimensionality, as the data volume grows exponentially with the number of parameters. Tensor decompositions circumvent this issue by exploiting low-rank structures in multidimensional data. Among these, the Hierarchical Tucker Decomposition (HTD), introduced by Hackbusch and Kühn [6] and Grasedyck [4], provides an efficient, data-sparse representation of high-dimensional tensors.

For a d -dimensional tensor $X \in \mathbb{R}^{n_1 \times n_2 \times \dots \times n_d}$, the HTD organizes the tensor modes in a binary dimension tree \mathcal{T}_d . Each node of \mathcal{T}_d corresponds to a subset of modes, with the root node representing the full set $\{1, \dots, d\}$. For every inner node t with children t_1 and t_2 , the relations $t = t_1 \cup t_2$ and $t_1 \cap t_2 = \emptyset$ hold.

Each node is associated with a basis matrix $U_t \in \mathbb{R}^{n_t \times r_t}$, where $n_t = \prod_{i \in t} n_i$. This matrix spans the column space of the matricization $X^{(t)}$, and $r_t = \text{rank}(U_t)$ is referred to as the hierarchical rank of node t .

If t_1 and t_2 are the children of t , a nestedness property allows the parent basis U_t to be constructed from the child bases $U_{t_1} \in \mathbb{R}^{n_{t_1} \times r_{t_1}}$, $U_{t_2} \in \mathbb{R}^{n_{t_2} \times r_{t_2}}$, and a transfer tensor $B_t \in \mathbb{R}^{r_{t_1} \times r_{t_2} \times r_t}$ as

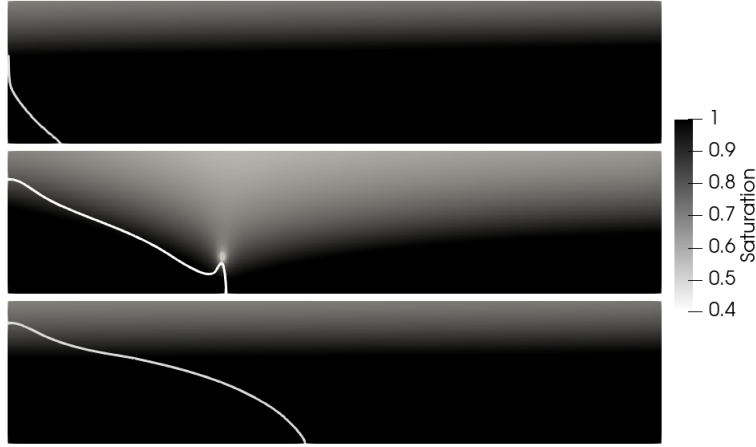


Fig. 2 Post-pumping seawater intrusion simulation. From top to bottom, the three images show simulation start ($t = 0$ s), pumping stop ($t = 135$ s) and post-pumping intrusion maximum ($t = 588$ s). The 0.5 salt concentration isoline is shown in white.

$$U_t = (U_{t_2} \otimes U_{t_1}) B_t^{(\{1,2\})},$$

where \otimes denotes the Kronecker product, and $B_t^{(\{1,2\})}$ is the matricization of B_t along its first two modes. At the root node, the corresponding transfer tensor encodes the information required to reconstruct the full vectorized tensor from the bases of its two children, denoted U_{root_1} and U_{root_2} :

$$\text{vec}(\mathbf{X}) = (U_{\text{root}_2} \otimes U_{\text{root}_1}) B_{\text{root}}^{(\{1,2\})}.$$

This recursive structure implies that a full tensor can be decomposed into a Hierarchical Tucker tensor composed of basis matrices U_t for all leaves and transfer tensors B_t for all inner nodes of \mathcal{T}_d . Consequently, the storage complexity of $\mathcal{O}(n^d)$ for a full tensor is reduced to $\mathcal{O}(dnr + dr^3)$ for a uniform mode size n and hierarchical rank r . However, it should be noted that the linear dependence on the number of dimensions d only holds if increasing d does not lead to a corresponding growth in the hierarchical rank r .

Sampling-based construction. In the context of the PDE model, we are interested in approximating the temporal evolution of the intrusion toe length for varying combinations of hydrogeological parameters α , n , K , ϕ and inflow Q (see Section 2). By $f(\alpha, n, K, \phi, Q, t)$ we denote the intrusion toe length from a specific parameter combination at time t . For each parameter, we introduce finite sets of values

$$\{\alpha_1, \alpha_2, \dots, \alpha_N\}, \{n_1, n_2, \dots, n_N\}, \dots, \{Q_1, Q_2, \dots, Q_N\},$$

Table 1 Selected parameter bounds and discretization used in the study.

Parameter	Lower bound	Upper bound	Discretization points
α [Pa^{-1}]	1.023E-4	1.023E-3	25
n [-]	1	3	25
K [m^2]	7.2E-10	14.4E-10	25
ϕ [-]	0.275	0.375	25
Q [$\text{m}\cdot\text{day}^{-1}$]	4.6	5.6	25
t [s]	0	5200	51

leading to a discrete parameter space of cardinality N^d . Together with a discrete set of time points $\{t_0, t_1, \dots, t_{M-1}\}$, this naturally defines a six-dimensional tensor $X \in \mathbb{R}^{N \times N \times N \times N \times N \times M}$ with entries

$$X[i_1, i_2, i_3, i_4, i_5, i_6] = f(\alpha_{i_1}, n_{i_2}, K_{i_3}, \phi_{i_4}, Q_{i_5}, t_{i_6}).$$

Evaluating the entire tensor is computationally infeasible, even for moderate values of N , as each parameter combination requires running a full simulation. To address this challenge, we employ the sampling-based approach proposed by Grasedyck et al. [5]. For each node t within the dimension tree, the algorithm identifies quasi-optimal row and column pivots (P_t and Q_t , respectively). This is achieved by applying cross-approximation to a randomly sampled submatrix of the node's matricization, with the objective of finding a submatrix of maximal volume. These pivots are subsequently utilized to construct the basis matrices for the leaf nodes and the transfer tensors for all inner nodes. The transfer tensors are derived by enforcing the nested coupling between a parent node t and its two children, t_1 and t_2 , expressed by the relation:

$$X^{(t)}[(p, q), j] = \sum_k \sum_l X^{(t_1)}[p, k] X^{(t_2)}[q, l] B_t[k, l, j],$$

where $p \in P_{t_1}$, $q \in P_{t_2}$, and $j \in Q_t$.

3 Results

Setup of the Parameter Study The conducted parameter study is based on the numerical model introduced in Section 2. Each parameter was varied within physically plausible bounds (see Table 1), guided by the ranges reported by Stoeckl et al. [11]. For all parameter combinations, the intrusion toe length was evaluated at the same set of non-uniformly spaced time points, with denser sampling in regions of expected rapid changes. The sampling procedure referred to in Section 2 was used to construct a corresponding Hierarchical Tucker tensor using a prescribed hierarchical rank of $r = 7$ at all nodes.

Table 2 Comparison between the evaluated simulation data, the constructed Hierarchical Tucker tensor, and its fully decompressed counterpart. Each evaluated sample corresponds to one time series of 51 data points. The HT tensor ($r=7$) compresses the data by nearly five orders of magnitude and effectively reconstructs about 1 806 time series for each computed simulation.

	Represented time series	Total data points	Memory [MB]
Decompressed full tensor	9 765 625	498 046 875	3984
Evaluated simulations N_{sim}	5 408	275 808	2.2
Constructed HT tensor	9 765 625	498 046 875	0.021

Sampling Efficiency and Approximation Accuracy. To evaluate the efficiency of the sampling-based HTD construction, we compare the number of computed simulations with the number of time series represented in the resulting Hierarchical Tucker tensor (see Table 2). The sampling algorithm evaluated $N_{sim} = 5\,408$ full simulations, corresponding to 275 808 time-series data points. From these, an HT tensor comprising 498 046 875 entries was constructed, effectively recovering about 1806 time series per computed sample. Despite encoding orders of magnitude more data, the resulting HT tensor requires about 100 times less memory than the raw sampling data. When compared to the explicitly reconstructed full tensor, its storage demand drops to merely 0.0005% of the total size. This efficiency results from the compact low-rank structure of the decomposition, where each simulation contributes only selectively to the basis matrices and transfer tensors.

Next, we evaluate the approximation accuracy of the constructed HT tensor using an independent test set of $n = 500$ simulations sampled using a uniform distribution from the discrete parameter space, excluding those used during tensor construction. Each simulation yields a time series $\mathbf{s}^{(i)} = s_1^{(i)}, \dots, s_{51}^{(i)}$, which is compared with the corresponding HT tensor prediction $\hat{\mathbf{s}}^{(i)} = \hat{s}_1^{(i)}, \dots, \hat{s}_{51}^{(i)}$ using the relative error defined as

$$\epsilon_{\text{rel}} = \sqrt{\sum_{i=1}^n \|\mathbf{s}^{(i)} - \hat{\mathbf{s}}^{(i)}\|_2^2} / \sqrt{\sum_{i=1}^n \|\mathbf{s}^{(i)}\|_2^2}.$$

To determine a suitable hierarchical rank, we evaluated HT tensors for $r = 5, \dots, 8$, requiring $N_{sim} = 4847, 5121, 5408, 5862$ full simulations, and yielding relative errors of 1.82%, 1.58%, 1.37% and 1.36%, respectively. The error decreases significantly up to $r = 7$, with further rank increases providing minimal improvement. As higher ranks require more extensive sampling during construction [5], we choose $r = 7$ as a compromise between accuracy and computational cost.

Application to the Intrusion Dynamics. We demonstrate two a-posteriori applications of the constructed HT tensor. First, Fig. 3a shows the variability of the parameter-dependent predicted intrusion toe lengths using quantile envelopes, which visualize the distribution of all encoded trajectories, alongside the experimentally measured time series from Ref. [11]. Second, we query the HT tensor for parameter combinations that most closely reproduce these observations (see Fig. 3 b); the iden-

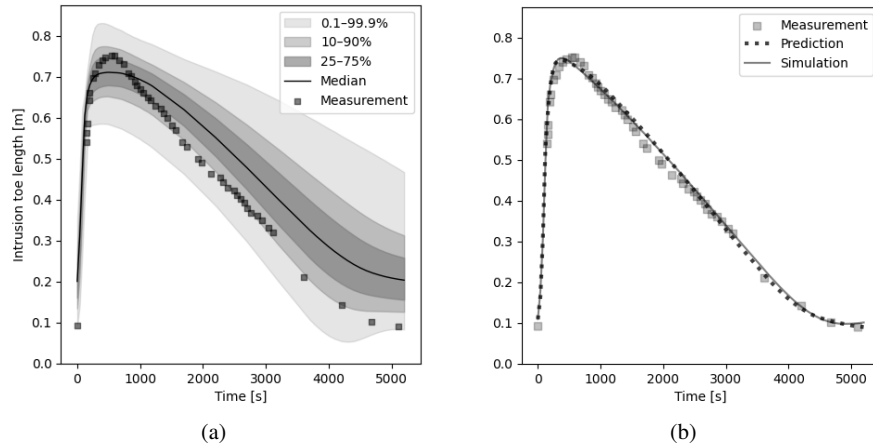


Fig. 3 Overview of the results from the HT tensor predictions: (a) Quantile envelopes of the intrusion toe evolution predicted by the HT tensor. The shaded regions reflect the variability within the parameter ranges defined in Table 1; (b) Time series from the HT tensor that best matches the experimentally measured intrusion toe lengths. The solid line shows the corresponding full simulation, included to verify the tensor prediction.

tified parameter combination is $\alpha = 7.16 \cdot 10^{-4} \text{Pa}^{-1}$, $n = 1.17$, $K = 7.2 \cdot 10^{-10} \text{m}^2$, $\phi = 0.338$, and $Q = 4.81 \text{m} \cdot \text{day}^{-1}$.

4 Discussion and Conclusion

Using a sampling-based approach building on the cross-approximation of matrices, we constructed a low-rank Hierarchical Tucker tensor that accurately reproduces the parameter-dependent dynamics of saltwater intrusion. On average, one fully computed simulation enabled the reconstruction of roughly 1 800 additional time series. This efficient representation addresses the curse of dimensionality that typically hinders exhaustive parameter studies. For the application of interest, a good agreement with experimental data was obtained. Possible future investigations include exploring the use of the proposed approach to generate robust initial guesses for gradient-based parameter optimization, thereby mitigating convergence to local minima, and assessing the potential of the constructed HT tensor to enable efficient parameter sensitivity analyses across the encoded space without additional computational cost.

Acknowledgements The authors thank Niklas Conen, Dmitry Logashenko, and all authors of [1] as their work has been instrumental for the study at hand. This work was supported by the German Federal Ministry for the Environment, Nature Conservation, Nuclear Safety and Consumer Protection (Grant No. 02E12012B).

References

1. Conen, N., Logashenko, D., Naegel, A., Schneider, A., Wittum, G., Zhao, H.: Simulation of haline flow in large scale unsaturated aquifers. *GEM Int. J. Geomath.* **17** (2026)
2. Fein, E., Schneider, A.: D³F – ein Programmpaket zur Modellierung von Dichteströmungen. GRS Report 139, Gesellschaft für Anlagen- und Reaktorsicherheit (GRS) mbH, Braunschweig (1998). URL <https://www.grs.de/sites/default/files/publications/GRS-139.pdf>
3. van Genuchten, M.T.: A closed-form equation for predicting the hydraulic conductivity of unsaturated soils. *Soil Sci. Soc. Am. J.* **44**(5), 892–898 (1980)
4. Grasedyck, L.: Hierarchical singular value decomposition of tensors. *SIAM J. Matrix Anal. Appl.* **31**, 2029–2054 (2010)
5. Grasedyck, L., Kriemann, R., Löbber, C., Nägel, A., Wittum, G., Xylouris, K.: Parallel tensor sampling in the hierarchical tucker format. *Comput. Vis. Sci.* **17**, 67–78 (2015)
6. Hackbusch, W., Kühn, S.: A new scheme for the tensor representation. *J. Fourier Anal. Appl.* **15**, 706–722 (2009)
7. Jiao, J., Post, V.: *Coastal Hydrogeology*. Cambridge University Press, Cambridge (2019)
8. Nägel, A., Deuffhard, P., Wittum, G.: Efficient stiff integration of Density Driven Flow Problems. ZIB Report 18-54, ZIB Berlin (2018)
9. Reiter, S., Vogel, A., Heppner, I., Rupp, M., Wittum, G.: A massively parallel geometric multigrid solver on hierarchically distributed grids. *Comput. Vis. Sci.* **16**(4), 151–164 (2013)
10. Scheidegger, A.E.: *The Physics of Flow Through Porous Media* (3rd Edition). University of Toronto Press (1974)
11. Stoeckl, L., Walther, M., Morgan, L.K.: Physical and numerical modelling of post-pumping seawater intrusion. *Geofluids* **2019**(1), 7191370 (2019)
12. Vogel, A., Reiter, S., Rupp, M., Nägel, A., Wittum, G.: Ug 4: A novel flexible software system for simulating pde based models on high performance computers. *Comput. Vis. Sci.* **16**(4), 165–179 (2013)