VI International Conference on Particle-based Methods – Fundamentals and Applications
PARTICLES 2019
E. Oñate, M. Bischoff, D.R.J. Owen, P. Wriggers & T. Zohdi (Eds)

MESHFREE SIMULATIONS FOR SOLUTION MINING PROCESSES

ISABEL MICHEL[1], TOBIAS SEIFARTH[1] AND JÖRG KUHNERT[1]

[1] Fraunhofer ITWM
Fraunhofer-Platz 1, 67663 Kaiserslautern, Germany
E-mail: contact@meshfree.eu and URL: https://www.meshfree.eu/

Key words: Solution Mining, Meshfree Method, GFDM, FPM, Lagrangian Formulation, Eulerian Formulation

Abstract. Experimental and field investigations for solution mining processes have improved intensely within the last years. Due to today's computing capacities, 3D simulations of potential salt solution caverns can further enhance the process understanding. They serve as a "virtual prototype" of a projected site and support planning in reasonable time. In this contribution, we present a meshfree generalized finite difference method based on a cloud of numerical points that is able to simulate solution mining processes on microscopic as well as macroscopic scales. Focusing on anticipated industrial requirements, Lagrangian and Eulerian formulations including an ALE-approach are considered.

1 INTRODUCTION

Classical simulation methods are meshbased. Moving geometry parts, free surfaces, phase boundaries, or large deformations are difficult to handle or require time-consuming re-meshing algorithms. This is not the case for meshfree methods. They show their advantages especially for these applications. The meshfree Finite Pointset Method (FPM) uses a generalized finite difference method (GFDM) on a cloud of numerical points. There are already successful applications of the method in CFD and continuum mechanics. Examples include water crossing of cars, water turbines, hydraulic valves, soil mechanics, and metal cutting [6, 8, 15, 19]. The current development of the simulation software MESHFREE has eliminated previous shortcomings concerning robust and scalable solutions of sparse, linear systems. MESHFREE combines the advantages of FPM and the fast linear solvers of SAMG [16]. In this contribution, we present its capabilities with respect to the simulation of solution mining processes on microscopic and macroscopic scales.

Originally, a Lagrangian formulation is used, i.e. the point cloud moves according to the flow velocity [6, 9]. Thus, there is an accurate and natural transport of physical information. The basic physical model consists of the conservation equations for mass, momentum, and energy. For solution mining processes, we extend it by the standard k-ε turbulence model and equations for the concentration of the occurring species (see Section 2). The GFDM specific numerics are derived in Section 3 with special focus on

the Lagrangian and Eulerian formulations. The microscopic simulations in Section 4 illustrate a procedure to determine the necessary effective model parameters of a macroscopic problem. For macroscopic simulations, the Lagrangian formulation leads to a significant restriction of the time step size due to the explicit movement of the point cloud. To enable simulations in reasonable time, the Eulerian formulation should be preferred [17]. Thereby, the point cloud is fixed and convective terms represent the transport of physical information. The necessary movement of the boundary of the salt cavern is implemented based on the solution rate. Close to the boundary, interior points are subject to an ALE-approach (Arbitrary Lagrangian-Eulerian) according to [4]. This procedure gives rise to covering the complete life cycle of a salt cavern by a meshfree simulation. We demonstrate the advantages of the Eulerian formulation for a simplified macroscopic example in Section 5, followed by conclusions in Section 6.

2 PHYSICAL MODEL

In this section, we describe the basic physical model and its extensions for solution mining processes. Furthermore, a specific model for the density of a solution is discussed.

2.1 Basic Equations

The basic physical model is given by the conservation equations of mass, momentum, and energy in Lagrangian formulation.

$$\begin{aligned} \frac{d\rho}{dt} + \rho \cdot \nabla^{\mathrm{T}} \mathbf{v} &= 0, \\ \frac{d}{dt}(\rho \cdot \mathbf{v}) + (\rho \cdot \mathbf{v}) \cdot \nabla^{\mathrm{T}} \mathbf{v} &= (\nabla^{\mathrm{T}} \mathbf{S})^{\mathrm{T}} - \nabla p + \rho \cdot \mathbf{g}, \\ \frac{d}{dt}(\rho \cdot E) + (\rho \cdot E) \cdot \nabla^{\mathrm{T}} \mathbf{v} &= \nabla^{\mathrm{T}}(\mathbf{S} \cdot \mathbf{v}) - \nabla^{\mathrm{T}}(p \cdot \mathbf{v}) + \rho \cdot \mathbf{g}^{\mathrm{T}} \cdot \mathbf{v} + \nabla^{\mathrm{T}}(\lambda \cdot \nabla T), \end{aligned} \tag{1}$$

where $\frac{d}{dt} = \frac{\partial}{\partial t} + \mathbf{v}^{\mathrm{T}} \nabla$ denotes the material derivative. Furthermore, we have: nabla operator $\nabla = (\frac{\partial}{\partial x}, \frac{\partial}{\partial y}, \frac{\partial}{\partial z})^{\mathrm{T}}$, density ρ, velocity $\mathbf{v} \in \mathbb{R}^3$, stress tensor $\mathbf{S} \in \mathbb{R}^{3\times 3}$ (deviatoric part, i.e. $\mathrm{tr}(\mathbf{S}) = 0$), pressure p, body forces $\mathbf{g} \in \mathbb{R}^3$, total energy $E = c_{\mathrm{v}} \cdot T + \frac{1}{2} \cdot (\mathbf{v}^{\mathrm{T}} \cdot \mathbf{v})$, heat capacity c_{v}, temperature T, and heat conductivity λ. As described in [6, 9], the stress tensor is split into its viscous and solid parts by $\mathbf{S} = \mathbf{S}_{\mathrm{visc}} + \mathbf{S}_{\mathrm{solid}}$. For simplicity, the solid part will be neglected for further analysis. The viscous part is defined by

$$\mathbf{S}_{\mathrm{visc}} = (\eta + \eta_{\mathrm{turb}}) \cdot \left(\nabla \mathbf{v}^{\mathrm{T}} + (\nabla \mathbf{v}^{\mathrm{T}})^{\mathrm{T}} - \frac{2}{3} \cdot (\nabla^{\mathrm{T}} \mathbf{v}) \cdot \mathbf{I} \right), \tag{2}$$

where $\mathbf{I} \in \mathbb{R}^{3\times 3}$ is the identity. To incorporate turbulent effects, the standard k-ε turbulence model is considered omitting fluctuating dilatation and source terms, see [14]:

$$\begin{aligned} \frac{dk}{dt} &= \frac{1}{\rho} \cdot \nabla^{\mathrm{T}} \left(\left(\eta + \frac{\eta_{\mathrm{turb}}}{\sigma_k} \right) \cdot \nabla k \right) - \varepsilon + \frac{1}{\rho} \cdot (P_{\mathrm{pr}} + P_{\mathrm{b}}), \\ \frac{d\varepsilon}{dt} &= \frac{1}{\rho} \cdot \nabla^{\mathrm{T}} \left(\left(\eta + \frac{\eta_{\mathrm{turb}}}{\sigma_\varepsilon} \right) \cdot \nabla \varepsilon \right) - C_{2\varepsilon} \cdot \frac{\varepsilon^2}{k} + \frac{1}{\rho} \cdot C_{1\varepsilon} \cdot \frac{\varepsilon}{k} \cdot (P_{\mathrm{pr}} + C_{3\varepsilon} \cdot P_{\mathrm{b}}), \end{aligned} \tag{3}$$

where η is the laminar and $\eta_{\text{turb}} = \rho \cdot C_\eta \cdot \frac{k^2}{\varepsilon}$ is the turbulent viscosity. The turbulent production rate is defined by $P_{\text{pr}} = \eta_{\text{turb}} \cdot \|\nabla \mathbf{v}^{\text{T}}\|_{\text{M}}^2$ with von Mises matrix norm $\|\cdot\|_{\text{M}}$. The turbulent buoyancy is given by $P_{\text{b}} = -\frac{1}{\rho} \cdot \frac{\eta_{\text{turb}}}{\text{Pr}_{\text{turb}}} \cdot \frac{\partial \rho}{\partial T} \cdot (\mathbf{g} \cdot \nabla T)$. For this model, well-established values for the constants are $\sigma_k = 1.0$, $\sigma_\varepsilon = 1.3$, $C_{1\varepsilon} = 1.44$, $C_{2\varepsilon} = 1.92$, $C_{3\varepsilon} = -0.33$, $C_\eta = 0.09$, and $\text{Pr}_{\text{turb}} = 0.85$ (turbulent Prandtl number). In the vicinity of walls, a logarithmic wall function is used. In order to simulate solution mining processes, the model above is extended by convection-diffusion-equations for concentration c_i of species $i = 1, \ldots, N$ with effective diffusion coefficient $D_{i,\text{eff}}$:

$$\frac{dc_i}{dt} + c_i \cdot \nabla^{\text{T}} \mathbf{v} = \nabla^{\text{T}} (D_{i,\text{eff}} \cdot \nabla c_i). \tag{4}$$

In the Eulerian formulation, the material derivative is replaced by its definition.

2.2 Modeling Density, Viscosity, and Heat Capacity

The general form of the equation of state is given by $\rho = \rho(T, c_1, \ldots, c_N)$, i.e. density depends on temperature and concentrations. Based on the formulation in [10, 13], the density of a solution of N species in water is given by

$$\rho_{\text{sol}} = \left(\frac{w_{\text{H}_2\text{O}}}{\rho_{\text{H}_2\text{O}}} + \sum_{i=1}^{N} \frac{w_i}{\rho_{\text{apparent},i}} \right)^{-1}, \tag{5}$$

where $w_{\text{H}_2\text{O}}$ and w_i are the mass fraction of water and species i, respectively. Additionally, $w_{\text{H}_2\text{O}} + \sum_{i=1}^{N} w_i = 1$ has to be satisfied. The density of water is determined by

$$\rho_{\text{H}_2\text{O}} = \frac{(((((A_1 \cdot T + A_2) \cdot T + A_3) \cdot T + A_4) \cdot T + A_5) \cdot T + A_6)}{1 + A_7 \cdot T}, \tag{6}$$

while the apparent density of species i is given by

$$\rho_{\text{apparent},i} = \frac{(C_0^* \cdot (1 - w_{\text{H}_2\text{O}}) + C_1^*) \cdot \exp\left(0.000001 \cdot (T + C_4^*)^2\right)}{(1 - w_{\text{H}_2\text{O}}) + C_2^* + C_3^* \cdot T} \tag{7}$$

with $A_1, \ldots, A_7$ and $C_0^*, \ldots, C_4^*$ according to [13]. The mass fractions w_i are defined by $w_i = \frac{c_i}{\sum_{i=1}^{N} c_i + \rho_{\text{H}_2\text{O}}}$. Similarly, $\eta_{\text{sol}}(T, c_1, \ldots, c_N)$ and $c_{\text{v,sol}}(T, c_1, \ldots, c_N)$ are modeled, see [10, 11, 12].

3 NUMERICS BASED ON GFDM

3.1 Point Cloud Management

The GFDM approach uses a discretization by a cloud of numerical points which compactly covers the computational domain. The density of the point cloud is given by a sufficiently smooth function $h = h(\mathbf{x}, t)$, the so-called interaction radius. There is a direct correlation between the point cloud resolution and the quality of the simulation results

(convergence in h of order 2). Details regarding the initial seeding and quality maintenance of the point cloud are found in [8, 9]. The latter is of utmost importance in the Lagrangian formulation as well as the ALE-approach since the movement of (parts of) the point cloud leads to accumulation or scattering of points which would reduce the quality of the numerical results.

3.2 Differential Operators

We use a specialized weighted moving least squares approach to determine numerical differential operators $\tilde{\partial}^0, \tilde{\partial}^x, \tilde{\partial}^y, \tilde{\partial}^z, \tilde{\Delta}$ for function approximation, x-,y-,z-derivative, and the Laplacian which are independent of the considered function, see [8]. The interaction radius defines the general neighbors of each point. During the least squares operator generation, the ones with the smallest distance to the considered point obtain the highest weight. Furthermore, the numerical operators are defined such that chosen discrete test functions and their derivatives are reproduced exactly. A common choice are monomials up to a certain order and the delta function.

3.3 Time Integration

A strong form discretization of the physical model (Section 2) can be provided based on the numerical differential operators and a chosen time integration scheme. For simplicity, the following considerations are based on first order time integration.

Starting with the Lagrangian formulation, equations (1) can be rewritten as

$$\begin{aligned} \frac{d\rho}{dt} &= -\rho \cdot \nabla^{\mathrm{T}} \mathbf{v}, \\ \frac{d\mathbf{v}}{dt} &= \frac{1}{\rho} \cdot (\nabla^{\mathrm{T}} \mathbf{S})^{\mathrm{T}} - \frac{1}{\rho} \cdot \nabla p + \mathbf{g}, \\ (\rho \cdot c_{\mathrm{v}}) \cdot \frac{dT}{dt} &= \nabla^{\mathrm{T}}(\mathbf{S} \cdot \mathbf{v}) - (\nabla^{\mathrm{T}} \mathbf{S}) \cdot \mathbf{v} - p \cdot \nabla^{\mathrm{T}} \mathbf{v} + \nabla^{\mathrm{T}}(\lambda \cdot \nabla T). \end{aligned} \tag{8}$$

Note that we use $\rho = \rho_{\mathrm{sol}}$ and $c_{\mathrm{v}} = c_{\mathrm{v,sol}}$ to improve readability. Together with equations (2)–(4), this is the starting point of the numerical discretization. The spatial derivatives are replaced by their least squares approximated counterparts. Denoting the future time level by $n+1$ and the current one by n with time step size $\Delta t = t^{n+1} - t^n$, the point cloud is moved according to a second order method by $\mathbf{x}^{n+1} = \mathbf{x}^n + \Delta t \cdot \mathbf{v}^n + \frac{\mathbf{v}^{\mathrm{n}} - \mathbf{v}^{\mathrm{n-1}}}{2 \cdot \Delta t_0} \cdot (\Delta t)^2$ with previous time step size Δt_0 wrt. time levels n and $n-1$, see [18]. A semi-implicit time integration for temperature T^{n+1} reads

$$(\mathbf{I}_T + \mathbf{D}_T) \cdot T^{n+1} = (\rho^n \cdot c_{\mathrm{v}}^n) \cdot T^n + f_T, \tag{9}$$

where $\mathbf{I}_T = \rho^n \cdot c_{\mathrm{v}}^n \cdot \mathbf{I}$, $\mathbf{D}_T = -\Delta t \cdot \tilde{\nabla}^{\mathrm{T}}(\lambda \cdot \tilde{\nabla})$, and $f_T = \Delta t \cdot (\tilde{\nabla}^{\mathrm{T}}(\mathbf{S}^n \cdot \mathbf{v}^n) - (\tilde{\nabla}^{\mathrm{T}} \mathbf{S}^n) \cdot \mathbf{v}^n - p^n \cdot \tilde{\nabla}^{\mathrm{T}} \mathbf{v}^n)$. Equation (9) forms a sparse linear system with unknowns T^{n+1} at each point of the point cloud. For the sake of clarity, the index of the point is omitted. Analogously, a semi-implicit time integration for concentrations c_i^{n+1}, $i = 1, \ldots, N$, can be formed:

$$(\mathbf{I}_{c_i} + \mathbf{D}_{c_i}) \cdot c_i^{n+1} = c_i^n, \tag{10}$$

where $\mathbf{I}_{c_i} = (\mathbf{I} + \Delta t \cdot \tilde{\nabla}^{\mathrm{T}}\mathbf{v}^n)$ and $\mathbf{D}_{c_i} = -\Delta t \cdot \tilde{\nabla}^{\mathrm{T}}(D_{i,\mathrm{eff}} \cdot \tilde{\nabla})$. The future density ρ^{n+1}, viscosities $\eta_{\mathrm{sol}}^{n+1}$ and $\hat{\eta}^{n+1} = \eta_{\mathrm{sol}}^{n+1} + \eta_{\mathrm{turb}}^{n}$, as well as heat capacity c_{v}^{n+1} can be determined according to Section 2.2.

Time integration of the first equation in (8) provides the targeted divergence of velocity $\tilde{\nabla}^{\mathrm{T}}\mathbf{v}^{n+1}$. To solve for $\mathbf{v}^{n+1}$ and p^{n+1} in an implicit time integration scheme, we use the penalty formulation introduced in [6, 9] which is independent of the Reynolds number. To this end, the pressure is split into its hydrostatic (body forces) and dynamic parts (movement of the fluid) by $p = p_{\mathrm{hyd}} + p_{\mathrm{dyn}}$. For the hydrostatic pressure, we have

$$\tilde{\nabla}^{\mathrm{T}}\left(\frac{1}{\rho^{n+1}} \cdot \tilde{\nabla} p_{\mathrm{hyd}}^{n+1}\right) = \tilde{\nabla}^{\mathrm{T}}\mathbf{g}. \tag{11}$$

Defining the preliminary pressure by $\hat{p} = p_{\mathrm{hyd}}^{n+1} + p_{\mathrm{dyn}}^{n}$, we obtain the following coupled velocity-pressure-system for preliminary velocity $\hat{\mathbf{v}}^{n+1}$ and correction pressure p_{corr}^{n+1}:

$$\left(\mathbf{I} - \frac{\Delta t}{\rho^{n+1}} \cdot \tilde{\psi}_{\hat{\eta}^{n+1}}^{n+1}\right) \cdot \hat{\mathbf{v}}^{n+1} + \frac{\Delta t}{\rho^{n+1}} \cdot \tilde{\nabla} p_{\mathrm{corr}}^{n+1} = \mathbf{v}^n - \frac{\Delta t}{\rho^{n+1}} \cdot \tilde{\nabla}\hat{p} + \Delta t \cdot \mathbf{g}, \tag{12}$$

$$\tilde{\nabla}^{\mathrm{T}}\left(\frac{\Delta t_{\mathrm{virt}}}{\rho^{n+1}} \cdot \tilde{\nabla} p_{\mathrm{corr}}^{n+1}\right) = \tilde{\nabla}^{\mathrm{T}}\hat{\mathbf{v}}^{n+1} - \tilde{\nabla}^{\mathrm{T}}\mathbf{v}^{n+1} \tag{13}$$

with $(\tilde{\psi}_{\hat{\eta}^{n+1}}^{n+1})^{\mathrm{T}} = \tilde{\nabla}^{\mathrm{T}}(\hat{\eta}^{n+1} \cdot \tilde{\nabla})(\hat{\mathbf{v}}^{n+1})^{\mathrm{T}} + (\tilde{\nabla}\hat{\eta}^{n+1})^{\mathrm{T}} \cdot (\tilde{\nabla}(\hat{\mathbf{v}}^{n+1})^{\mathrm{T}})^{\mathrm{T}} + \frac{\hat{\eta}^{n+1}}{3} \cdot (\tilde{\nabla}(\tilde{\nabla}^{\mathrm{T}}\hat{\mathbf{v}}^{n+1}))^{\mathrm{T}} - \frac{2}{3} \cdot (\tilde{\nabla}^{\mathrm{T}}\hat{\mathbf{v}}^{n+1}) \cdot (\tilde{\nabla}\hat{\eta}^{n+1})^{\mathrm{T}}$ and $\Delta t_{\mathrm{virt}} = A_{\mathrm{virt}} \cdot \Delta t$, $0 \leq A_{\mathrm{virt}} \leq 1$. If $A_{\mathrm{virt}} = 1$, the scheme corresponds to an implicit Chorin projection, see [2]. Theoretically, choosing $A_{\mathrm{virt}} = 0$ would give the exact solution. However, the linear system is ill-conditioned and can not be solved in most cases. For $0.001 \leq A_{\mathrm{virt}} \leq 0.1$, conditioning of the linear system is sufficiently good. Furthermore, the resulting preliminary velocity features a divergence which is very close to the targeted one. The update of the velocity and the dynamic pressure are given by $\mathbf{v}^{n+1} = \hat{\mathbf{v}}^{n+1} - \frac{\Delta t_{\mathrm{virt}}}{\rho^{n+1}} \cdot \tilde{\nabla} p_{\mathrm{corr}}^{n+1}$ and $p_{\mathrm{dyn}}^{n+1} = p_{\mathrm{dyn}}^{n} + p_{\mathrm{corr}}^{n+1}$. The updated stress tensor $\mathbf{S}^{n+1}$ is determined according to (2).

For the k-ε turbulence model, we derive a singularity formulation from (3):

$$\begin{aligned}
\frac{d}{dt}\left(\frac{k}{\varepsilon}\right) =& (C_{2\varepsilon} - 1) + C_\eta \cdot (1 - C_{1\varepsilon}) \cdot \|\tilde{\nabla}\mathbf{v}^{\mathrm{T}}\|_{\mathrm{M}}^2 \cdot \left(\frac{k}{\varepsilon}\right)^2 \\
&+ \frac{C_\eta \cdot (C_{1\varepsilon} \cdot C_{3\varepsilon} - 1)}{\rho \cdot \mathrm{Pr}_{\mathrm{turb}}} \cdot \frac{\partial \rho}{\partial T} \cdot (\mathbf{g} \cdot \tilde{\nabla} T) \cdot \left(\frac{k}{\varepsilon}\right)^2 + \frac{1}{\rho} \cdot \tilde{\Delta}_{\eta^*}\left(\frac{k}{\varepsilon}\right), \\
\frac{d}{dt}\left(\frac{\varepsilon}{k}\right) =& (1 - C_{2\varepsilon}) \cdot \left(\frac{\varepsilon}{k}\right)^2 + C_\eta \cdot (C_{1\varepsilon} - 1) \cdot \|\tilde{\nabla}\mathbf{v}^{\mathrm{T}}\|_{\mathrm{M}}^2 \\
&+ \frac{C_\eta \cdot (1 - C_{1\varepsilon} \cdot C_{3\varepsilon})}{\rho \cdot \mathrm{Pr}_{\mathrm{turb}}} \cdot \frac{\partial \rho}{\partial T} \cdot (\mathbf{g} \cdot \tilde{\nabla} T) + \frac{1}{\rho} \cdot \tilde{\Delta}_{\eta^*}\left(\frac{\varepsilon}{k}\right),
\end{aligned} \tag{14}$$

where $\tilde{\Delta}_{\eta^*}\left(\frac{k}{\varepsilon}\right) = \frac{\varepsilon \cdot \tilde{\Delta}_{\eta_k} k - k \cdot \tilde{\Delta}_{\eta_\varepsilon} \varepsilon}{\varepsilon^2}$ and $\tilde{\Delta}_{\eta^*}\left(\frac{\varepsilon}{k}\right) = \frac{k \cdot \tilde{\Delta}_{\eta_\varepsilon} \varepsilon - \varepsilon \cdot \tilde{\Delta}_{\eta_k} k}{k^2}$ with $\tilde{\Delta}_{\eta_k} = \tilde{\nabla}^{\mathrm{T}}\left(\left(\eta + \frac{\eta_{\mathrm{turb}}}{\sigma_k}\right) \cdot \tilde{\nabla}\right)$ and $\tilde{\Delta}_{\eta_\varepsilon} = \tilde{\nabla}^{\mathrm{T}}\left(\left(\eta + \frac{\eta_{\mathrm{turb}}}{\sigma_\varepsilon}\right) \cdot \tilde{\nabla}\right)$, respectively. If $k, \varepsilon > 0$ for all $t^n \leq t \leq t^{n+1}$, numerical mean values can be determined from (14): $\frac{k}{\varepsilon}\big|_{\mathrm{m}} = \frac{1}{\Delta t}\int_{t^n}^{t^{n+1}} \frac{d}{dt}\left(\frac{k}{\varepsilon}\right) dt$, $\frac{\varepsilon}{k}\big|_{\mathrm{m}} =$

$\frac{1}{\Delta t}\int_{t^n}^{t^{n+1}} \frac{d}{dt}\left(\frac{\varepsilon}{k}\right)\, dt$. We use the mean values to avoid singularities in the discretized k-ε turbulence model.

$$\frac{dk}{dt} = \frac{\tilde{\Delta}_{\eta_k} k}{\rho} - \left.\frac{\varepsilon}{k}\right|_{\mathrm{m}} \cdot k + C_\eta \cdot \left(\|\tilde{\nabla}\mathbf{v}^{\mathrm{T}}\|_{\mathrm{M}}^2 - \frac{1}{\rho \cdot \mathrm{Pr}_{\mathrm{turb}}} \cdot \frac{\partial \rho}{\partial T} \cdot (\mathbf{g} \cdot \tilde{\nabla} T) \right) \cdot \left.\frac{k}{\varepsilon}\right|_{\mathrm{m}} \cdot k, \tag{15}$$

$$\frac{d\varepsilon}{dt} = \frac{\tilde{\Delta}_{\eta_\varepsilon} \varepsilon}{\rho} - C_{2\varepsilon} \cdot \left.\frac{\varepsilon}{k}\right|_{\mathrm{m}} \cdot \varepsilon + C_{1\varepsilon} \cdot C_\eta \cdot \left(\|\tilde{\nabla}\mathbf{v}^{\mathrm{T}}\|_{\mathrm{M}}^2 - \frac{C_{3\varepsilon}}{\rho \cdot \mathrm{Pr}_{\mathrm{turb}}} \cdot \frac{\partial \rho}{\partial T} \cdot (\mathbf{g} \cdot \tilde{\nabla} T) \right) \cdot \left.\frac{k}{\varepsilon}\right|_{\mathrm{m}} \cdot \varepsilon.$$

Consequently, a fully implicit time integration scheme for k^{n+1} reads

$$\begin{aligned} & k^{n+1} - \frac{\Delta t \cdot \tilde{\Delta}_{\eta_k} k^{n+1}}{\rho} + \Delta t \cdot \left.\frac{\varepsilon}{k}\right|_{\mathrm{m}} \cdot k^{n+1} \\ & - \Delta t \cdot C_\eta \cdot \left(\|(\tilde{\nabla}\mathbf{v}^{n+1})^{\mathrm{T}}\|_{\mathrm{M}}^2 - \frac{1}{\rho \cdot \mathrm{Pr}_{\mathrm{turb}}} \cdot \frac{\partial \rho}{\partial T} \cdot (\mathbf{g} \cdot \tilde{\nabla} T^{n+1}) \right) \cdot \left.\frac{k}{\varepsilon}\right|_{\mathrm{m}} \cdot k^{n+1} = k^n. \end{aligned} \tag{16}$$

Analogously, we have

$$\begin{aligned} & \varepsilon^{n+1} - \frac{\Delta t \cdot \tilde{\Delta}_{\eta_\varepsilon} \varepsilon^{n+1}}{\rho} + \Delta t \cdot C_{2\varepsilon} \cdot \left.\frac{\varepsilon}{k}\right|_{\mathrm{m}} \cdot \varepsilon^{n+1} \\ & - \Delta t \cdot C_{1\varepsilon} \cdot C_\eta \cdot \left(\|(\tilde{\nabla}\mathbf{v}^{n+1})^{\mathrm{T}}\|_{\mathrm{M}}^2 - \frac{C_{3\varepsilon}}{\rho \cdot \mathrm{Pr}_{\mathrm{turb}}} \cdot \frac{\partial \rho}{\partial T} \cdot (\mathbf{g} \cdot \tilde{\nabla} T^{n+1}) \right) \cdot \left.\frac{k}{\varepsilon}\right|_{\mathrm{m}} \cdot \varepsilon^{n+1} = \varepsilon^n. \end{aligned} \tag{17}$$

The mean values are determined analytically. This is illustrated in detail for $\left.\frac{k}{\varepsilon}\right|_{\mathrm{m}}$. Assuming that the diffusion term $\frac{1}{\rho} \cdot \tilde{\Delta}_{\eta^*}\left(\frac{k}{\varepsilon}\right)$ is negligible as well as defining $x = \frac{k}{\varepsilon}$, $a = C_{2\varepsilon} - 1$, and $b = C_\eta \cdot \left((C_{1\varepsilon} - 1) \cdot \|\tilde{\nabla}\mathbf{v}^{\mathrm{T}}\|_{\mathrm{M}}^2 + \frac{(1 - C_{1\varepsilon} \cdot C_{3\varepsilon})}{\rho \cdot \mathrm{Pr}_{\mathrm{turb}}} \cdot \frac{\partial \rho}{\partial T} \cdot (\mathbf{g} \cdot \tilde{\nabla} T) \right)$, we can rewrite equation (14) as $\frac{dx}{dt} = a - b \cdot x^2$. Thus, we obtain

$$x^{n+1} = \begin{cases} \sqrt{\frac{a}{b}} \cdot \tanh\left(\Delta t \cdot \sqrt{a \cdot b} + \operatorname{arctanh}\left(\sqrt{\frac{b}{a}} \cdot x^n \right) \right), & x^n < \sqrt{\frac{a}{b}} \\ \sqrt{\frac{a}{b}}, & x^n = \sqrt{\frac{a}{b}} \\ \sqrt{\frac{a}{b}} \cdot \coth\left(\Delta t \cdot \sqrt{a \cdot b} + \operatorname{arccoth}\left(\sqrt{\frac{b}{a}} \cdot x^n \right) \right), & x^n > \sqrt{\frac{a}{b}} \end{cases}. \tag{18}$$

The updated turbulent viscosity is determined by $\eta_{\mathrm{turb}}^{n+1} = \rho^{n+1} \cdot C_\eta \cdot \frac{(k^{n+1})^2}{\varepsilon^{n+1}}$.

In case of the Eulerian formulation, [17] shows that a second order time integration scheme should be applied to numerically solve transport terms of the form $\mathbf{v}^{\mathrm{T}}\nabla$. For this purpose, the SDIRK2 method is proposed (see [1]). It features the same stability properties as an implicit Euler time integration scheme. Furthermore, an upwind discretization by means of a MUSCL reconstruction with a Superbee limiter is used. For example, the coupled velocity-pressure-system is solved by the following two-step procedure:

$$\begin{aligned} \left(\mathbf{I}_{\hat{\mathbf{v}}^{n+\alpha}} - \frac{\alpha \cdot \Delta t}{\rho^{n+\alpha}} \tilde{\psi}_{\hat{\eta}^{n+\alpha}}^{n+\alpha} \right) \cdot \hat{\mathbf{v}}^{n+\alpha} + \frac{\alpha \cdot \Delta t}{\rho^{n+\alpha}} \cdot \tilde{\nabla} p_{\mathrm{corr}}^{n+\alpha} &= \mathbf{v}^n - \frac{\alpha \cdot \Delta t}{\rho^{n+\alpha}} \cdot \tilde{\nabla}\hat{p} + \alpha \cdot \Delta t \cdot \mathbf{g}, \\ \tilde{\nabla}^{\mathrm{T}} \left(\frac{\Delta t_{\mathrm{virt}}}{\rho^{n+\alpha}} \cdot \tilde{\nabla} p_{\mathrm{corr}}^{n+\alpha} \right) &= \tilde{\nabla}^{\mathrm{T}} \hat{\mathbf{v}}^{n+\alpha} - \tilde{\nabla}^{\mathrm{T}} \mathbf{v}^{n+\alpha} \end{aligned} \tag{19}$$

with $\mathbf{I}_{\hat{\mathbf{v}}^{n+\alpha}} = (\mathbf{I} + (\widetilde{\mathbf{v}^{\mathrm{T}}\nabla})\hat{\mathbf{v}}^{n+\alpha})$ and $(\tilde{\psi}^{n+\alpha}_{\hat{\eta}^{n+\alpha}})^{\mathrm{T}} = \tilde{\nabla}^{\mathrm{T}}(\hat{\eta}^{n+\alpha} \cdot \tilde{\nabla})(\hat{\mathbf{v}}^{n+\alpha})^{\mathrm{T}} + (\tilde{\nabla}\hat{\eta}^{n+\alpha})^{\mathrm{T}} \cdot (\tilde{\nabla}(\hat{\mathbf{v}}^{n+\alpha})^{\mathrm{T}})^{\mathrm{T}} + \frac{\hat{\eta}^{n+\alpha}}{3} \cdot (\tilde{\nabla}(\tilde{\nabla}^{\mathrm{T}}\hat{\mathbf{v}}^{n+\alpha}))^{\mathrm{T}} - \frac{2}{3} \cdot (\tilde{\nabla}^{\mathrm{T}}\hat{\mathbf{v}}^{n+\alpha}) \cdot (\tilde{\nabla}\hat{\eta}^{n+\alpha})^{\mathrm{T}}$ and, subsequently,

$$\begin{aligned} \hat{\mathbf{v}}^{n+1} - \Delta t \cdot \alpha \cdot \mathbf{V}(\hat{\mathbf{v}}^{n+1}, p^{n+1}_{\mathrm{corr}}) &= \mathbf{v}^n + \Delta t \cdot (1-\alpha) \cdot \mathbf{V}(\hat{\mathbf{v}}^{n+\alpha}, p^{n+\alpha}_{\mathrm{corr}}), \\ \tilde{\nabla}^{\mathrm{T}} \left(\frac{\Delta t_{\mathrm{virt}}}{\rho^{n+1}} \cdot \tilde{\nabla} p^{n+1}_{\mathrm{corr}} \right) &= \tilde{\nabla}^{\mathrm{T}}\hat{\mathbf{v}}^{n+1} - \tilde{\nabla}^{\mathrm{T}}\mathbf{v}^{n+1} \end{aligned} \tag{20}$$

with $\mathbf{V}(\hat{\mathbf{v}}^{n+1}, p^{n+1}_{\mathrm{corr}}) = -\frac{1}{\rho^{n+1}} \cdot (\widetilde{\mathbf{v}^{\mathrm{T}}\nabla})\hat{\mathbf{v}}^{n+1} + \frac{1}{\rho^{n+1}} \cdot \tilde{\psi}^{n+1}_{\hat{\eta}^{n+1}} - \frac{1}{\rho^{n+1}} \cdot \tilde{\nabla}\hat{p}^{n+1} + \mathbf{g}$ and $\mathbf{V}(\hat{\mathbf{v}}^{n+\alpha}, p^{n+\alpha}_{\mathrm{corr}}) = \frac{\hat{\mathbf{v}}^{n+\alpha} - \mathbf{v}^n}{\alpha \cdot \Delta t}$ are solved for $\alpha = 1 - \frac{\sqrt{2}}{2}$. Density and viscosity for the intermediate step can for instance be determined by linear interpolation between time level n and $n+1$.

4 MICROSCOPIC SCALE

In this subsection, we present a general method based on the Lagrangian formulation to identify the effective parameters that are necessary for macroscopic scale simulations. These are effective diffusion coefficient and boundary condition between water and surrounding species. For simplicity, we restrict the following illustration to sodium chloride. Please note that the procedure can directly be transferred to any other species.

4.1 Setup

We consider a cylinder with diameter of 5m and height of 10m which is initially filled with pure water, i.e. $c_{\mathrm{NaCl}}(t=0) = 0$. During the simulation, the temperature is fixed to $T_0 = 20\,^{\circ}\mathrm{C}$. The roof of the cylinder acts as inexhaustible supply of sodium chloride which is modeled by applying a Dirichlet condition with saturation concentration $c^{\mathrm{s}}_{\mathrm{NaCl}} = c^{\mathrm{s}}_{\mathrm{NaCl}}(T_0) = 357 \frac{\mathrm{kg}}{\mathrm{m}^3}$. For the hull of the cylinder, a homogeneous Neumann condition is applied. Aiming at a quasi-stationary profile, the bottom of the cylinder models an outflow boundary. In the interior, we solve $\frac{\partial c_{\mathrm{NaCl}}}{\partial t} = \nabla^{\mathrm{T}}(D_{\mathrm{micro}} \cdot \nabla c_{\mathrm{NaCl}})$, where $D_{\mathrm{micro}} = D_{\mathrm{laminar}} + D_{\mathrm{turb}}$. The laminar diffusion coefficient for sodium chloride is given by $D_{\mathrm{laminar}} = 1.611 \cdot 10^{-9} \frac{\mathrm{m}^2}{\mathrm{s}}$ (see [3]). For the turbulent part, we have $D_{\mathrm{turb}} = C_\eta \cdot \frac{k^2}{\varepsilon}$. Standard boundary conditions (Dirichlet and Neumann) are prescribed for velocity, pressure, and the turbulent quantities. The simulation runs until a quasi-stationary state is reached.

4.2 Evaluation Strategy

In order to determine the effective quantities, the cylinder is split in z-direction into equal sub-cylinders SC_j, $j = 1, \ldots, J$, which are used to estimate the mass flow. The planes between the sub-cylinders are denoted by HP_j, $j = 1, \ldots, J-1$. Analogously to the heat flow, the mass flow of sodium chloride is given by $\frac{dm}{dt} = -D_{\mathrm{NaCl,eff}} \cdot \frac{\partial \bar{c}_{\mathrm{NaCl}}}{\partial \mathbf{n}}$, where $\bar{c}_{\mathrm{NaCl}}$ is the mean concentration. The mass flow and the mean concentration in sub-cylinder SC_j are determined by

$$\frac{dm}{dt}(SC_j) = \frac{\int_{SC_j} c_{\mathrm{NaCl}} \cdot v_3 \, dV_{SC_j}}{\int_{SC_j} 1 \, dV_{SC_j}}, \quad \bar{c}_{\mathrm{NaCl}}(SC_j) = \frac{\int_{SC_j} c_{\mathrm{NaCl}} \, dV_{SC_j}}{\int_{SC_j} 1 \, dV_{SC_j}}. \tag{21}$$

Based on the mean concentration in a sub-cylinder SC_j, we can approximate its normal derivative wrt. the help plane HP_j. This yields the effective diffusion coefficients

$$D_{\mathrm{NaCl,eff}}(SC_j|HP_j) = -\frac{\frac{dm}{dt}(SC_j)}{\frac{\partial \bar{c}_{\mathrm{NaCl}}}{\partial \mathbf{n}}\big|_{HP_j}}, \quad j = 1, \ldots, J-1. \tag{22}$$

As soon as a quasi-stationary state is reached, the values for the different j tend to the desired effective diffusion coefficient for a macroscopic setup (resolution is of the order of the height of the sub-cylinders). To accommodate the "quasi-stationary" character of a simulation, we use a time-averaged effective diffusion coefficient wrt. a small time interval.

In the macroscopic simulation, we use the Robin boundary condition

$$D_{\mathrm{NaCl,eff}} \cdot \frac{\partial c_{\mathrm{NaCl}}}{\partial \mathbf{n}} = \gamma_{\mathrm{NaCl,eff}} \cdot (c_{\mathrm{NaCl}} - c^{\mathrm{s}}_{\mathrm{NaCl}}), \tag{23}$$

where the effective transition coefficient $\gamma_{\mathrm{NaCl,eff}}$ can be derived from the microscopic setup similarly to the effective diffusion coefficient $D_{\mathrm{NaCl,eff}}$:

$$\gamma_{\mathrm{NaCl,eff}}(SC_j) = \frac{\frac{dm}{dt}(SC_j)}{\bar{c}_{\mathrm{NaCl}}(SC_j) - c^{\mathrm{s}}_{\mathrm{NaCl}}}, \quad j = 1, \ldots, J-1. \tag{24}$$

If the quasi-stationary state is reached, the (time-averaged) sub-cylinder specific values tend to the desired value of the effective transition coefficient. With the help of $\gamma_{\mathrm{NaCl,eff}}$, we can define the solution rate of sodium chloride for given temperature T_0 by

$$R_{\mathrm{NaCl}}(T_0) = \gamma_{\mathrm{NaCl,eff}}(c_{\mathrm{NaCl}} - c^{\mathrm{s}}_{\mathrm{NaCl}}). \tag{25}$$

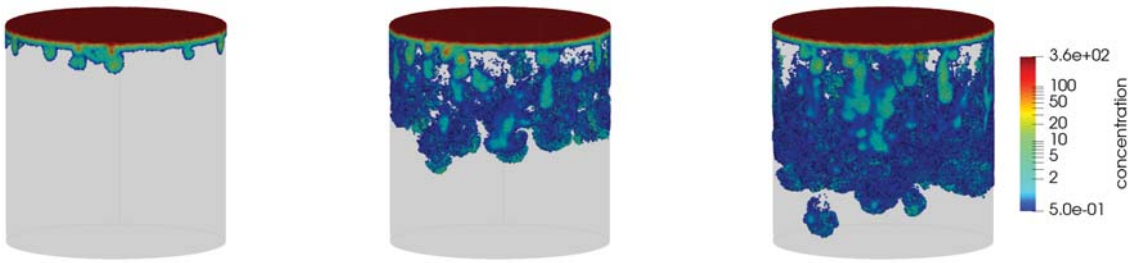
4.3 Numerical Results

The resolution (interaction radius h) in the macroscopic simulation in Section 5 is of the order of meters. Thus, we choose $J = 10$ in the microscopic setup which yields sub-cylinders of height 1m. We consider five levels of resolution with interaction radii $h = 0.30\mathrm{m}, 0.25\mathrm{m}, 0.20\mathrm{m}, 0.19\mathrm{m}, 0.18\mathrm{m}$. The evolution of the concentration for the highest resolution in the time interval $[0\mathrm{s}, 100\mathrm{s}]$ is illustrated in Figure 1. As expected, the flow is characterized by viscous fingering.

Decreasing the interaction radius h, leads to convergence of the estimated effective diffusion and transition coefficients towards $D_{\mathrm{NaCl,eff}} = 0.1\frac{\mathrm{m}^2}{\mathrm{s}}$ and $\gamma_{\mathrm{NaCl,eff}} = 0.000042\frac{\mathrm{m}}{\mathrm{s}}$, respectively (see Table 1). According to this, we obtain a solution rate of $R_{\mathrm{NaCl}}(20\,^{\circ}\mathrm{C}) = -0.0150\frac{\mathrm{kg}}{\mathrm{m}^2\cdot\mathrm{s}}$. Compared to the solution rate of $-0.0488\frac{\mathrm{kg}}{\mathrm{m}^2\cdot\mathrm{s}}$ for $T_0 = 23\,^{\circ}\mathrm{C}$ determined in [7] on crystal level, the estimated solution rate is of the correct order of magnitude.

5 MACROSCOPIC SCALE

The effective parameters determined in Section 4 are used in the macroscopic setup below which models a simplified solution mining process. Both the Lagrangian as well as the Eulerian formulation are evaluated, see [17].



(a) $t = 10\text{s}$ **(b)** $t = 55\text{s}$ **(c)** $t = 100\text{s}$

Figure 1: Evolution of the microscopic simulation for $h = 0.18\text{m}$ – concentration.

Table 1: Estimated effective diffusion and transition coefficient, as well as solution rate.

interaction radius	$D_{\text{NaCl,eff}}$	$\gamma_{\text{NaCl,eff}}$	$R_{\text{NaCl}}(20\,^\circ\text{C})$
$h = 0.30\text{m}$	$0.1515\frac{\text{m}^2}{\text{s}}$	$0.000184\frac{\text{m}}{\text{s}}$	$-0.0657\,\frac{\text{kg}}{\text{m}^2\cdot\text{s}}$
$h = 0.25\text{m}$	$0.1558\frac{\text{m}^2}{\text{s}}$	$0.000141\frac{\text{m}}{\text{s}}$	$-0.0504\,\frac{\text{kg}}{\text{m}^2\cdot\text{s}}$
$h = 0.20\text{m}$	$0.1096\frac{\text{m}^2}{\text{s}}$	$0.000074\frac{\text{m}}{\text{s}}$	$-0.0264\,\frac{\text{kg}}{\text{m}^2\cdot\text{s}}$
$h = 0.19\text{m}$	$0.1028\frac{\text{m}^2}{\text{s}}$	$0.000050\frac{\text{m}}{\text{s}}$	$-0.0179\,\frac{\text{kg}}{\text{m}^2\cdot\text{s}}$
$h = 0.18\text{m}$	$0.0999\frac{\text{m}^2}{\text{s}}$	$0.000042\frac{\text{m}}{\text{s}}$	$-0.0150\,\frac{\text{kg}}{\text{m}^2\cdot\text{s}}$

5.1 Setup

We are interested in the geometrical evolution of a salt cavern. The initial geometry is given by a small cavern filled with pure water which is surrounded by sodium chloride, see Figure 2. The dimensions of the initial cavern are approximately: width of 90m, height of 50m, and depth of 26m. The sodium chloride deposit is limited in extension which corresponds to impermeable surrounding rock. The pipe on the left side acts as an inlet of fresh water with $|\mathbf{v}_{\text{in}}| = 1\frac{\text{m}}{\text{s}}$, whereas the pipe on the right side acts as the outlet.

In reality, the maximum diameter of the pipes is of the order of 1m. Hence, the

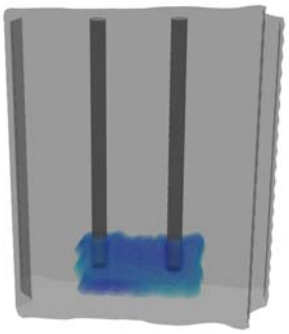


Figure 2: Macroscopic simulation setup – initial geometry, see [17].

resolution of the point cloud close to the inlet and the outlet has to be of the order of 0.1m to ensure accurate results. In case of the Lagrangian formulation, this would lead to an extremely small time step size compared to the desired simulation time of several years due to the CFL-condition $\Delta t_{\text{Lag}} \leq CFL_{\text{Lag}} \cdot \frac{h_{\min}}{|\mathbf{v}|}$. Stable results are achieved for $CFL_{\text{Lag}} = 0.15$. Since $|\mathbf{v}| \geq |\mathbf{v}_{\text{in}}|$ and $h_{\min} = \mathcal{O}(0.1\text{m})$, we obtain $\Delta t_{\text{Lag}} = \mathcal{O}(0.1\text{s})$. In order to allow for a comparison of Lagrangian and Eulerian formulation, we consider pipes of diameter 12m and constant interaction radius of $h = 4$m. Furthermore, we fix the temperature to $T_0 = 20\,^{\circ}\text{C}$ and, subsequently, obtain the corresponding saturation concentration $c^{\text{s}}_{\text{NaCl}} = 357\frac{\text{kg}}{\text{m}^3}$. For simplicity, the following linear relations for density and viscosity of the solution are used as described in [17]: $\rho(c_{\text{NaCl}}) \approx (0.56 \cdot c_{\text{NaCl}} + 1000)\frac{\text{kg}}{\text{m}^3}$, $\eta(c_{\text{NaCl}}) \approx (1.96 \cdot 10^{-6} \cdot c_{\text{NaCl}} + 10^{-3})\frac{\text{Pa}}{\text{s}}$. We use $D_{\text{NaCl,eff}} = 0.1\frac{\text{m}^2}{\text{s}}$ and $\gamma_{\text{NaCl,eff}} = 0.00005\frac{\text{m}}{\text{s}}$.

5.2 Movement of the Boundary

The movement of the boundary of the cavern can be defined by the Stefan condition $\rho v^{\star} = \gamma_{\text{NaCl,eff}}(c_{\text{NaCl}} - c^{\text{s}}_{\text{NaCl}})$, see [5]. This yields $v^{\star} = \frac{\gamma_{\text{NaCl,eff}}}{\rho} \cdot (c_{\text{NaCl}} - c^{\text{s}}_{\text{NaCl}})$ and, consequently, a movement of the boundary in normal direction $\mathbf{n}$ with velocity $\mathbf{v}_{\text{boundary}} = v^{\star} \cdot \mathbf{n}$. To speed up computation, a time lapse procedure can be applied which introduces an additional factor A in the definition of $v^{\star}$ which is limited by certain stability conditions. For details, the reader is referred to [17].

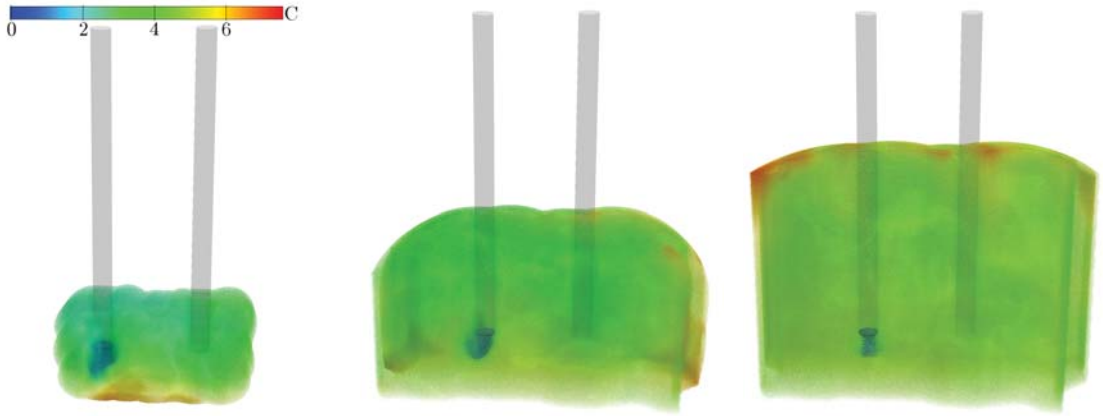
Due to the movement of the boundary, interior points close to this boundary have to move in the Eulerian formulation also. For this purpose, the ALE-approach presented in [4] is used. Based on current and future position of an affected interior point, the translational velocity $\mathbf{v}_{\text{trans}} = \frac{\mathbf{x}^{n+1} - \mathbf{x}^{n}}{\Delta t}$ is determined. Due to the explicit movement of these points, the convection terms in the numerical model in Eulerian form in Section 3.3 must refer to the relative velocity $\mathbf{v} - \mathbf{v}_{\text{trans}}$ instead of $\mathbf{v}$. Furthermore, this introduces a CFL-condition of the form $\Delta t_{\text{ALE}} \leq CFL_{\text{ALE}} \cdot \frac{h_{\min}}{v^{\star}}$. It depends on the boundary velocity $v^{\star} = \mathcal{O}(0.01\frac{\text{m}}{\text{s}})$ which is considerably smaller than the flow velocity, cf. [17]. Furthermore, $h_{\min}$ is subject to the desired resolution at the moving boundary. At the inlet and the outlet, a coarse resolution is sufficient in this case.

5.3 Numerical Results

Figure 3 illustrates the evolution of the salt cavern in the Eulerian formulation according to $\text{C} = c_{\text{NaCl}}$. The need for an ALE-concept is emphasized by the difference in time steps to cover a simulation time of 7200s: in the Lagrangian formulation, 22915 time steps are necessary, whereas only 936 time steps suffice in the Eulerian formulation. This corresponds to a speed-up of approximately 25 in computation time.

6 CONCLUSIONS

In this contribution, we presented the capabilities of the simulation software MESHFREE regarding solution mining processes on different scales. The developed generalized finite difference approach on a point cloud enables the use of a Lagrangian as well as an Eulerian formulation. On the microscopic scale, we described a procedure to deter-



(a) $t = 1537$s **(b)** $t = 2446$s **(c)** $t = 3228$s

Figure 3: Evolution of the macroscopic simulation for $h = 4$m – concentration, see [17].

mine effective diffusion and transition coefficients for an arbitrary species based on the Lagrangian formulation by considering sodium chloride as numerical example. The resulting effective parameters are then used in macroscopic simulations. A comparison of the simulation results for the Lagrangian and the Eulerian formulation (extended by an ALE-approach) illustrate the advantages of the latter one. Aiming at a simulation time of several years, the forecast computation time for a simulation based on the Lagrangian formulation would be of the order of years. In contrast to that, the flexibility of the Eulerian formulation regarding the resolution of the point cloud (local refinement only at the moving boundary) enables meshfree simulations in reasonable time – especially in terms of real applications.

REFERENCES

[1] Alexander, R. *Diagonally Implicit Runge-Kutta Methods for Stiff O.D.E.'s.* SIAM J. Numer. Anal. (1977) **14(6)**:1006–1021.

[2] Chorin, A.J. *Numerical solution of the Navier-Stokes equations.* Math. Comput. (1968) **22**:745–762.

[3] Flury, M. and Gimmi, T. *Solute Diffusion.* In: Dane, J.H. and Topp, G.C. (eds) Methods of Soil Analysis, Part 4, Physical Methods (2002) pp. 1323–1351.

[4] Hirt, C.W. and Amsden, A.A. and Cook, J.L. *An Arbitrary Lagrangian-Eulerian Computing Method for All Flow Speeds.* J. Comput. Phys. (1974) **14(3)**:227–253.

[5] Javierre-Perez, E. *Literature Study: Numerical Methods for Solving Stefan Problems.* Delft University of Technology (2003).

[6] Jefferies, A. and Kuhnert, J. and Aschenbrenner, L. and Giffhorn, U. *Finite Pointset Method for the Simulation of a Vehicle Travelling Through a Body of Water.* In:

Griebel, M. and Schweitzer, M.A. (eds) Meshfree Methods for Partial Differential Equations VII. LNCSE, Vol. 100, Springer, Cham (2015) pp. 205–211.

[7] Karsten, O. *Lösungsgeschwindigkeit von Natriumchlorid, Kaliumchlorid und Kieserit in Wasser und in wässerigen Lösungen*. ZAAC (1954) **276(5-6)**:247–266.

[8] Kuhnert, J. and Michel, I. and Mack, R. *Fluid Structure Interaction (FSI) in the MESHFREE Finite Pointset Method (FPM): Theory and Applications*. In: Griebel, M. and Schweitzer, M.A. (eds) Meshfree Methods for Partial Differential Equations IX, IWMMPDE2017. LNCSE, Vol. 129, Springer, Cham (2019) pp. 73–92.

[9] Kuhnert, J. *Meshfree numerical schemes for time dependent problems in fluid and continuum mechanics*. In: Sundar, S. (ed) Advances in PDE Modeling and Computation, Ane Books, New Delhi (2014) pp. 119–136.

[10] Laliberté, M. *A Model for Calculating the Heat Capacity of Aqueous Solutions, with Updated Density and Viscosity Data*. J. Chem. Eng. Data (2009) **54**:1725–1760.

[11] Laliberté, M. *Model for Calculating the Viscosity of Aqueous Solutions*. J. Chem. Eng. Data (2007) **52**:1507–1508.

[12] Laliberté, M. *Model for Calculating the Viscosity of Aqueous Solutions*. J. Chem. Eng. Data (2007) **52**:321–335.

[13] Laliberté, M. and Cooper W.E. *Model for Calculating the Density of Aqueous Electrolyte Solutions*. J. Chem. Eng. Data (2004) **49**:1141–1151.

[14] Launder, B.E. and Spalding, D.B. *The numerical computation of turbulent flows*. Comput. Methods. Appl. Mech. Eng. (1974) **3**:269–289.

[15] Michel, I. and Bathaeian, S.M.I. and Kuhnert, J. and Kolymbas, D. and Chen, C.-H. and Polymerou, I. and Vrettos, C. and Becker, A. *Meshfree generalized finite difference methods in soil mechanics – part II: numerical results*. Int. J. Geomath. (2017) **8(2)**:191–217.

[16] Nick, F. and Plum, H.-J. and Kuhnert, J. *Parallel Detection of Subsystems in Linear Systems Arising in the MESHFREE Finite Pointset Method*. In: Griebel, M. and Schweitzer, M.A. (eds) Meshfree Methods for Partial Differential Equations IX, IWMMPDE2017. LNCSE, Vol. 129, Springer, Cham (2019) pp. 93–115.

[17] Seifarth, T. *Numerische Algorithmen für gitterfreie Methoden zur Lösung von Transportproblemen*. PhD Thesis, University of Kassel, Fraunhofer Verlag (2018).

[18] Suchde, P. and Kuhnert, J. *Point cloud movement for fully Lagrangian meshfree methods*. J. Comput. Appl. Math. (2018) **340**:89–100.

[19] Uhlmann, E. and Gerstenberger, R. and Kuhnert, J. *Cutting simulation with the meshfree Finite Pointset Method*. Procedia CIRP (2013) **8**:391–396.