Universität Stuttgart

SimTech

C. Grüninger[a] · T. Fetzer[a] · B. Flemisch[a] · R. Helmig[a]

Coupling DuMuX and DUNE-PDELab to investigate evaporation at the interface between Darcy and Navier-Stokes flow

Stuttgart, November 2017

[a] Institute for Modelling Hydraulic and Environmental Systems, University of Stuttgart,
Pfaffenwaldring 61, 70569 Stuttgart/ Germany
{christoph.grueninger, thomas.fetzer, bernd.flemisch, rainer.helmig}@iws.uni-stuttgart.de
www.iws.uni-stuttgart.de

Abstract An implementation of a coupled Navier-Stokes/Darcy model based on different DUNE discretization modules is presented. The Darcy model is taken from DuMu$^\text{x}$, the Navier-Stokes model is implemented on top of DUNE-PDELab, and the coupling is done with help of DUNE-MultiDomain together with some project-specific auxiliary code.
The Navier-Stokes model features one fluid phase, the Darcy model two fluid phases. Each fluid phase may be composed of two components, in addition, non-isothermal processes are considered. The coupling between free and porous-medium flow uses a sharp interface between both subdomains and conserves mass, momentum, and energy by accounting for the corresponding fluxes across the interface.
A cell-centered finite volume method (FVM) is combined with a marker and cell (MAC) scheme. It solves the coupled problem in one monolithic system using a Newton method and a direct linear solver. Numerical results demonstrate the basic functioning and a lab-scale reference application.

Keywords Coupling · Darcy flow · Free flow · Evaporation · Drying · Multi-phase multi-component

Preprint Series **Issue No. 2017 - 1**
Stuttgart Research Centre for Simulation Technology (SRC SimTech)

SimTech – Cluster of Excellence
Pfaffenwaldring 5a
70569 Stuttgart

publications@simtech.uni-stuttgart.de
www.simtech.uni-stuttgart.de

1 Introduction

Evaporation processes from a porous medium to a free-flow region, like shown in figure 1a, are common in technical, environmental and biological applications: Various industrial drying applications [10], cooling of rocket engines [7], pre-lens tear film on a contact lens [32], subsurface nuclear waste repositories [24], and salt precipitation close to the soil surface in arid regions [21]. Further, quantifying evaporation rates of soil moisture to the atmosphere has a huge impact on climate models [29].

Notwithstanding its practical relevance, there are no user-friendly, ready-to-use software packages; available are either research codes (as the one presented in this work) or customized models for commercial solvers which require expert knowledge. We are aware of the following related works, which have the complete stack of physics needed to describe evaporation from a porous medium: The work from Dahmen et al. [7] examines gas cooling by evaporation from porous ceramics. They compare three-dimensional simulation results, obtained from an own implementation, with lab experiments. But they do not consider the vapor transport in the free flow. Defraeye et al. [11] couple two different pieces of software to describe the drying of concrete. The coupling is sequential, the Darcy subdomain is solved first. It is assumed that the flux is outwards with respect to the Darcy subdomain, such that multiple iterations between the subdomains are not necessary. Baber et al. [1] and Fetzer et al. [14] present a fully coupled implementation which uses the box scheme for both the Navier-Stokes and the Darcy subdomain. They lack a stable description of the free flow, especially for higher velocities and they are restricted to two-dimensional problems.

Our work is a successor of this approach intended to eliminate these weak points. We did not want to develop everything from scratch, nor can our concept be implemented with different pieces of software, because they are in general not prepared for a coupling. Our approach is to combine different modules, based on the Dune framework, to have a common ground and avoid the coupling of two black boxes of software.

Many researchers examined the coupling of a plain (Navier-)Stokes/Darcy flow. In a review article Discacciati and Quarteroni [12] cite more than a dozen mathematical analyses and several experimental and application works. Simulations of evaporation in porous media are complicated [10] because they require multiphase, multiscale and multiphysics modeling. To model evaporation in a porous medium, flow and transport in the liquid and gaseous phase must be considered. The water table drops and the vapor diffuses through the pores towards the interface, see figure 1b. In the free-flow region, the vapor is transported away. The phase change from liquid to gas invokes a cooling effect which retards further evaporation.

We use the Navier-Stokes equations to describe the free flow, the Darcy equation for the porous-medium flow, and a coupling across a sharp interface invoking the Beavers-Joseph condition [33], see figure 1c. We do discretize the Navier-Stokes equation with a marker and cell (MAC) scheme, the Darcy equation with a finite-volume method (FVM), and couple both. This is an established approach: Das et al. [8] describe a coupling in a three-dimensional, single-phase, single-component and isothermal setting. Rybak et al. [30] use these schemes to couple the isothermal Richards equation to the Stokes equation. Masson et al. [24] introduce an isothermal, two-component coupling for investigating evaporation in galleries from deep geological nuclear waste storage sites.

This work is organized as follows: In section 2 we describe the physical and mathematical models in the subdomains and for the coupling. Section 3 contains the used discretization. Section 4 presents the implementation details, especially the part for the coupling. In section 5 we show the results of numerical tests to exhibit the effective operation of our implementation. We conclude and give an outlook in the final section 6.

2 Model description

In this work, we use a two-domain concept, joining a porous-medium and a free-flow domain with a sharp interface Γ in-between, see figure 1a. Quantities of the gas phase are indicated by an index g, the ones of the liquid phase by l; variables belonging to the free-flow region have, if necessary, the superscript ff, similar pm for the porous medium region. All symbols and their physical meaning are compiled in the table 1. Note that several material properties like density ϱ or viscosity μ are

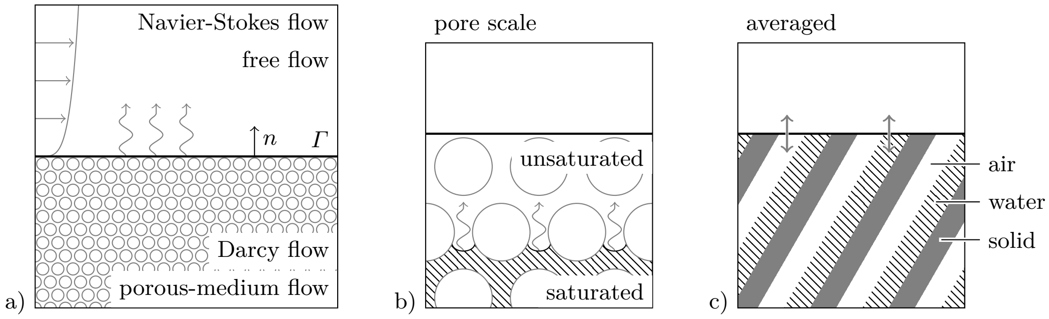


Fig. 1 (a) General problem of coupled Navier-Stokes/Darcy flow (b) situation with a water table below the interface and vapor diffusion in the gas phase (c) conceptual model

Table 1 Nomenclature

c_s	specific heat capacity of solid matrix	$\mathrm{J/kg\,K}$
D	diffusion coefficient	$\mathrm{m^2/s}$
g	gravitational acceleration vector	$\mathrm{m/s^2}$
h	specific enthalpy	$\mathrm{J/kg}$
j	diffusive fluxes	$\mathrm{kg/m^2s}$
K	intrinsic permeability	$\mathrm{m^2}$
$k_{\mathrm{r},\alpha}$	relative permeability, calculated after Van Genuchten	-
M	molar mass	$\mathrm{kg/mol}$
n	outer normal of porous medium subdomain regarding $\mathit{\Gamma}$	-
p	pressure	$\mathrm{kg/m\,s^2}$
p_c	capillary pressure, calculated after Van Genuchten	$\mathrm{kg/m\,s^2}$
S	saturation	-
t	time	s
t_i	tangential of $\mathit{\Gamma}$ with $t_{i,\mathit{\Gamma}} \perp t_{j,\mathit{\Gamma}}$ for $i \neq j$	-
T	temperature	K
u	specific internal energy	$\mathrm{J/kg}$
v	velocity	$\mathrm{m/s}$
x	space coordinate $x = (x_1, x_2, \dots)^\intercal$	m
x_mol	component mole fraction	-
X	component mass fraction	-
V, ∂V	element from discretization, edge between two elements	-
α	phase index	
α_BJ	Beavers-Joseph coefficient	-
$\mathit{\Gamma}$	interface between porous-medium and free-flow subdomain	-
κ	component index	
λ	heat conductivity	$\mathrm{W/mK}$
μ	dynamic viscosity	$\mathrm{kg/m\,s}$
ϱ	density	$\mathrm{kg/m^3}$
ϱ_mol	molar density	$\mathrm{mol/m^3}$
τ	tortuosity factor	-
ϕ	porosity	-
ff, pm	free flow, porous medium subdomain superscript	
a, w	air, water component superscript	
g, l, s	gas, liquid, solid phase subscript	

functions of the primary variables pressure p, vapor mass fraction X_g^w and temperature T. We assume Newtonian fluids, Fickian binary diffusion and local thermodynamic equilibrium within the subdomains [18].

2.1 Free flow

In the free-flow domain, we have a gaseous phase consisting of air and water vapor. Its velocities v and pressure p are the primary variables. We use the Navier-Stokes equations

$$\frac{\partial}{\partial t}\left(\varrho_\mathrm{g} v_\mathrm{g}\right) + \operatorname{div}\left(\varrho_\mathrm{g} v_\mathrm{g} v_\mathrm{g}^\intercal\right) - \operatorname{div}\left(\mu_\mathrm{g} \operatorname{grad} v_\mathrm{g}\right) + \operatorname{grad} p_\mathrm{g} = 0 \tag{1}$$

and the mass balance equation

$$\frac{\partial}{\partial t}\varrho_\mathrm{g} + \operatorname{div}\left(\varrho_\mathrm{g} v_\mathrm{g}\right) = 0. \tag{2}$$

For the free flow we neglect gravitational forces because the gas density is small, just like the resulting forces.

The vapor in the gaseous phase is considered as a component $\kappa = \mathrm{w}$. The mass balance equation for the vapor mass fraction X_g^w is

$$\frac{\partial}{\partial t}\left(\varrho_\mathrm{g} X_\mathrm{g}^\mathrm{w}\right) + \operatorname{div}\left(\varrho_\mathrm{g} X_\mathrm{g}^\mathrm{w} v_\mathrm{g}\right) + \operatorname{div} j_\mathrm{ff}^\mathrm{w} = 0. \tag{3}$$

The choice of $\kappa = \mathrm{w}$ over $\kappa = \mathrm{a}$ is arbitrary in the case of binary diffusion. The diffusive component fluxes are given by $j_\mathrm{ff}^\kappa = -D_\mathrm{g} \varrho_\mathrm{mol,g} M^\kappa \operatorname{grad} x_\mathrm{mol,g}^\kappa$.

We include the temperature with the energy balance equation

$$\frac{\partial}{\partial t}\left(\varrho_\mathrm{g} u_\mathrm{g}\right) + \operatorname{div}\left(\varrho_\mathrm{g} h_\mathrm{g} v_\mathrm{g}\right) + \sum_{\kappa \in \{\mathrm{a,w}\}} \operatorname{div}\left(h_\mathrm{g}^\kappa j_\mathrm{ff}^\kappa\right) - \operatorname{div}\left(\lambda_\mathrm{g} \operatorname{grad} T\right) = 0 \tag{4}$$

in a way that includes the enthalpy transport by diffusive component fluxes as the third term in the equation [14].

2.2 Porous-medium flow

Darcy's law describes flow through a porous medium. The law can be derived by averaging the Navier-Stokes equations inside the pores over a representative elementary volume (REV), see figure 1c, and can be stated as

$$v_\alpha = -\frac{k_{\mathrm{r},\alpha}}{\mu_\alpha} K \left(\operatorname{grad} p_\alpha - \varrho_\alpha g\right). \tag{5}$$

The velocity becomes a secondary variable, proportional to the pressure gradient. We use an extension for multiphase flow [18]. Our porous-medium model consists of a rigid solid matrix, the two mobile phases α liquid (l) and gas (g) and the two components κ air (a) and water (w).
The mass balance equation for a component reads as

$$\sum_{\alpha \in \{\mathrm{g,l}\}} \left(\phi \frac{\partial}{\partial t}\left(\varrho_\alpha S_\alpha X_\alpha^\kappa\right) + \operatorname{div}\left(\varrho_\alpha X_\alpha^\kappa v_\alpha\right) + \operatorname{div} j_{\mathrm{pm},\alpha}^\kappa\right) = 0 \quad \text{for } \kappa \in \{\mathrm{a, w}\} \tag{6}$$

with the diffusive fluxes $j_{\mathrm{pm},\alpha}^\kappa = -D_{\mathrm{pm},\alpha} \varrho_{\mathrm{mol},\alpha} M^\kappa \operatorname{grad} x_{\mathrm{mol},\alpha}^\kappa$. The effective diffusion coefficient $D_{\mathrm{pm},\alpha}$ is described by the tortuosity factor of the porous medium $\tau = 1/2$ and the molecular diffusion coefficient D_α as $D_{\mathrm{pm},\alpha} = D_\alpha \tau \phi S_\alpha$. The liquid phase is the wetting phase, the gaseous one is non-wetting. By definition, phase saturations S_α and component mass fractions X_α^κ sum to one, $S_\mathrm{g} + S_\mathrm{l} = 1$ and $X_\alpha^\mathrm{a} + X_\alpha^\mathrm{w} = 1$. The capillary pressure p_c is a relation between the two phase pressures, $p_\mathrm{c} = p_\mathrm{g} - p_\mathrm{l}$. The Van Genuchten model expresses p_c as a function of saturation. We use Dalton's law to calculate the partitioning processes. With these closing conditions we can express two unknowns in both mass balance equations. We choose a p_g–S_l formulation with a variable switch in case of a phase change and use p_g–X_l^a or p_g–X_g^w respectively. For the implementation, we reformulate (6) to the total mass balance equation by summing over $k \in \{\mathrm{a,w}\}$ and discretize this in addition to the water component balance.

As we assume local thermodynamic equilibrium, all phases have the same temperature T and the thermal equilibrium can be described by a single energy equation

$$\sum_{\alpha \in \{\mathrm{g,l}\}} \phi \frac{\partial}{\partial t}\left(\varrho_\alpha S_\alpha u_\alpha\right) + (1-\phi)\frac{\partial}{\partial t}\left(\varrho_\mathrm{s} c_\mathrm{s} T\right) + \sum_{\alpha \in \{\mathrm{g,l}\}} \operatorname{div}\left(\varrho_\alpha h_\alpha v_\alpha\right) - \operatorname{div}\left(\lambda_\mathrm{pm} \operatorname{grad} T\right) = 0. \tag{7}$$

Similar to the diffusion coefficient, the effective thermal conductivity λ_pm accounts for properties of soil, water and air. It is a function of the phase saturations.

2.3 Interface and Coupling Conditions

We couple both subdomains across a sharp interface by describing the interface fluxes for respective quantities. The set of equations for the coupling conditions originates from Layton et al. [23], Mosthaf et al. [25], and Fetzer et al. [14]. At the interface, we assume a local thermodynamic equilibrium. We include the exchange of mass and momentum with the continuity of total mass fluxes

$$\left(\varrho_{\mathrm{g}} v_{\mathrm{g}}\right)^{\mathrm{ff}} \cdot n = \left(\varrho_{\mathrm{g}} v_{\mathrm{g}} + \varrho_{\mathrm{l}} v_{\mathrm{l}}\right)^{\mathrm{pm}} \cdot n \quad \text{on } \mathit{\Gamma} \tag{8}$$

and the continuity of the normal stresses

$$p_{\mathrm{g}}^{\mathrm{ff}} + n^{\intercal} \left(\varrho_{\mathrm{g}} v_{\mathrm{g}} v_{\mathrm{g}}^{\intercal} - \mu_{\mathrm{g}} \operatorname{grad} v_{\mathrm{g}}\right)^{\mathrm{ff}} n = p_{\mathrm{g}}^{\mathrm{pm}} \quad \text{on } \mathit{\Gamma} \tag{9}$$

where n is the outer unit normal across the interface $\mathit{\Gamma}$ with respect to the porous medium. Diffusive fluxes cancel out because of the binary diffusion. The boundary condition for the momenta tangential to the interface is the Beavers-Joseph-Saffman condition [31]

$$\left(\alpha_{\mathrm{BJ}} v_{\mathrm{g}}^{\mathrm{ff}} - \sqrt{k} \operatorname{grad} v_{\mathrm{g}}^{\mathrm{ff}}\, n\right) \cdot t_i = 0 \quad \text{on } \mathit{\Gamma} \tag{10}$$

for all linear independent tangential vectors t_i. The condition is derived from a laminar single-phase flow. We expect a negligible influence for our applications, because the tangential movement within the porous medium has little effect on the evaporation.
For the phase components, here vapor, we use the continuity of vapor mass fraction

$$X_{\mathrm{g}}^{\mathrm{w,ff}} = X_{\mathrm{g}}^{\mathrm{w,pm}} \tag{11}$$

and the continuity of vapor mass fluxes

$$\left(\varrho_{\mathrm{g}} X_{\mathrm{g}}^{\mathrm{w}} v_{\mathrm{g}} + j_{\mathrm{ff}}^{\mathrm{w}}\right)^{\mathrm{ff}} \cdot n = \sum_{\alpha \in \{\mathrm{g,l}\}} \left(\varrho_{\alpha} X_{\alpha}^{\mathrm{w}} v_{\alpha} + j_{\mathrm{pm},\alpha}^{\mathrm{w}}\right)^{\mathrm{pm}} \cdot n. \tag{12}$$

Further, we employ the continuity of temperature

$$T^{\mathrm{ff}} = T^{\mathrm{pm}} \tag{13}$$

and the continuity of energy fluxes

$$\left(\varrho_{\mathrm{g}} h_{\mathrm{g}} v_{\mathrm{g}} + h_{\mathrm{g}}^{\mathrm{w}} j_{\mathrm{ff}}^{\mathrm{w}} + h_{\mathrm{g}}^{\mathrm{a}} j_{\mathrm{ff}}^{\mathrm{a}} - \lambda_{\mathrm{g}} \operatorname{grad} T\right)^{\mathrm{ff}} \cdot n = \left(\varrho_{\mathrm{g}} h_{\mathrm{g}} v_{\mathrm{g}} + \varrho_{\mathrm{l}} h_{\mathrm{l}} v_{\mathrm{l}} - \lambda_{\mathrm{pm}} \operatorname{grad} T\right)^{\mathrm{pm}} \cdot n. \tag{14}$$

3 Discretization

In this work, we use cell-centered finite volume methods (FVM) and a two-point flux approximation to discretize all porous-medium balance equations (6), (7), the free flow component transport (3) and the heat equation (4). The marker and cell (MAC) scheme from Harlow and Welch [17] is similar to the FVM but the degrees of freedom for the velocities are moved half an element to the faces of a control volume for the pressure, cf. figure 2a. We use a regular rectangular grid to facilitate calculating the staggered grid for the velocities. The two subdomain grids match at the free-flow/porous-medium-flow interface. All primary variables, except the velocities, are colocated at the element centers, see figure 2b.

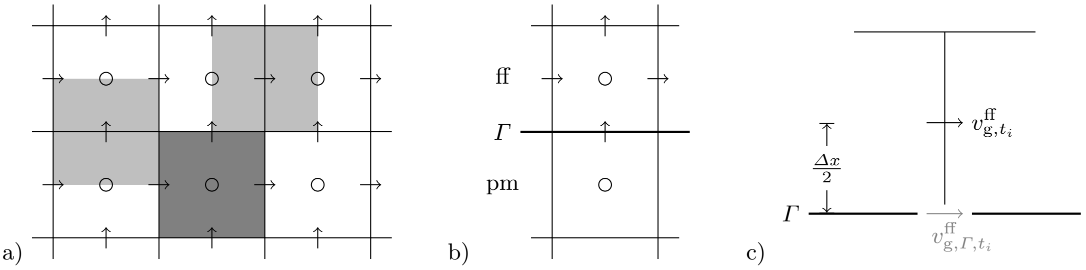

a)
ff
Γ
pm
b)
$\frac{\Delta x}{2}$
$v^{\text{ff}}_{\text{g},t_i}$
Γ
$v^{\text{ff}}_{\text{g},\Gamma,t_i}$
c)

Fig. 2 (a) Location of degrees of freedom in MAC scheme: arrows indicate velocities, circles all other primary variables (b) Coupling of a free-flow and a Darcy cell (c) Position of variables and helper variables for the Beavers-Joseph-Saffman condition

3.1 Coupling

For the Navier-Stokes equations (1), the two coupling conditions (9), (10) are realized in form of boundary conditions. In particular, the continuity of the normal stresses (9) is a Neumann boundary condition for the normal momentum. The Neumann flux value is the porous medium pressure integrated over the interface. As p^{pm}_{g} at the interface is not available from the discretization, we use the pressure from the adjacent porous-medium element center. The Beavers-Joseph-Saffman condition (10) is a Robin-type boundary condition for the tangential momentum of (1). It is implemented as a solution-dependent Neumann boundary condition. The i-th tangential velocity at the interface $v^{\text{ff}}_{\text{g}} \cdot t_i \big|_{\Gamma} =: v^{\text{ff}}_{\text{g},\Gamma,t_i}$ is no primary variable, cf. figure 2c. It occurs in (10) and we use it for approximating the derivative of the tangential velocity in the normal direction $\operatorname{grad}\left(v^{\text{ff}}_{\text{g}}\right) n \cdot t_i \approx \left(v^{\text{ff}}_{\text{g},t_i} - v^{\text{ff}}_{\text{g},\Gamma,t_i}\right) \frac{2}{\Delta x}$, which is part of the same equation that can be solved for $v^{\text{ff}}_{\text{g},\Gamma,t_i}$. Using this result, we can calculate the approximation of $\operatorname{grad}\left(v^{\text{ff}}_{\text{g}}\right) n \cdot t_i$ which is the Neumann boundary condition for the tangential momentum, because $\varrho_{\text{g}} v^{\text{ff}}_{\text{g}} v^{\text{ff},\intercal}_{\text{g}}$ vanishes in the proximity of the interface. For the mass balance equation (2), no coupling condition is required and the mass flux is simply calculated on the corresponding finite volume face.

For the total mass balance equation in the porous medium subdomain, the continuity of the total mass fluxes (8) is used as a Neumann boundary condition. The Neumann flux value is the free-flow momentum $\varrho^{\text{ff}}_{\text{g}} v^{\text{ff}}_{\text{g}} \cdot n$.

In principle, we would like to apply a standard two-point flux approximation with harmonic averaging in order to discretize (13) and the diffusive parts of (14). The latter proves to be difficult due to the different terms occurring on both sides of (14), especially since direct relationships between the individual terms are unknown. Therefore, we evaluate the free-flow part of the equation to calculate the interface flux, which is imposed as a Neumann boundary condition for both energy equations (4) and (7). The occurring partial derivatives of temperature in the normal direction are approximated with a finite difference including both values located at the element centers: $\operatorname{grad} T \cdot n \approx \left(T^{\text{ff}} - T^{\text{pm}}\right) \frac{1}{\Delta x}$. We discretize the equations (11), (12), which model the vapor mass fluxes, in an analogous manner.

The time is discretized with an implicit Euler scheme. Both subdomains and and the coupling use a common time step with adaptive step sizes. The resulting coupled system of non-linear equations is linearized with a Newton method in each time step, the resulting system of linear equations is solved with a direct solver in each Newton step.

4 Implementation

The open-source simulator DuMu$^{\text{x}}$ [15] in version 2.10 [19] is the basis of our implementation. We rely especially on the adaptive time stepping, the material laws and the non-isothermal two-phase, two-component Darcy model discretized by a cell-centered finite-volume method. It contains all physical laws and a variable switch to account for the vanishing and appearance of phases [6]. DuMu$^{\text{x}}$ is based on DUNE, the Distributed and Unified Numerics Environment [2, 3] which provides grids, linear solvers, local functions and defines interfaces for these. We use DUNE in the version

2.4.1 [5]. DUNE-PDELab is a discretization module with the goal to ease implementing discretization schemes [4]. We use PDELab for our implementation of the MAC scheme, mainly because it supports mixed degree of freedom handling for element and intersection centers. DUNE-MultiDomainGrid and DUNE-MultiDomain [27] extend PDELab for multi-physics simulations. DUNE-MultiDomainGrid can subdivide a given grid and handle the subdomains like grids of their own. Additionally, it offers an iterator over the coupling interface. DUNE-MultiDomain uses this structure and applies so-called local operators to each subdomain and the coupling interface. The local operator for the coupling interface follows the description in the last section. The free-flow subdomain local operator is the MAC scheme together with a FVM for component mass and energy transport, implemented with PDELab, see subsection 4.1. The Darcy subdomain local operator is the DuMu$^\text{x}$-based FVM code. As DuMu$^\text{x}$ local operators are incompatible to PDELab and MultiDomain, we provide a wrapper to tunnel the residual and Jacobian computation to PDELab, see subsection 4.2. The system is linearized with Newton's method. UMFPack [9] from SuiteSparse 4.4 solves the resulting monolithic sparse linear system with a direct solver. The remaining adjustments needed for our implementation of the coupling are covered in subsection 4.3. In the following, file names in parentheses help to find according passages in our code.

4.1 MAC local operator

The MAC scheme can be interpreted as a cell-centered finite volume method for the pressure combined with cell-centered finite volume methods on staggered grids, shifted by half an element, for the velocities. Discretizing equation (2) with the pressure as primary variable is straight forward with a cell-centered FVM. Avoiding multiple staggered grids, we piece the contributions from the momentum balance equation (1) together as volume or skeleton terms on the primal grid. We follow the naming from PDELab, where volume terms denote element contributions and skeleton terms are contributions from the vertex between two elements. Volume terms can only depend on degrees of freedom from the same element, skeleton terms are restricted to degrees of freedom from the two adjacent elements.

For a discretized domain, equation (1) can be formulated as

$$\int_{\tilde{V}} \frac{\partial}{\partial t} \left(\varrho_\mathrm{g} v_\mathrm{g}\right) + \int_{\partial\tilde{V}} \left(\varrho_\mathrm{g} v_\mathrm{g} v_\mathrm{g}^\intercal\right) \cdot n_{\partial\tilde{V}} - \int_{\partial\tilde{V}} \left(\mu_\mathrm{g} \operatorname{grad} v_\mathrm{g}\right) \cdot n_{\partial\tilde{V}} + \int_{\partial\tilde{V}} p_\mathrm{g} n_{\partial\tilde{V}} = 0, \tag{15}$$

when it is discretized by a finite volume scheme for an element $\tilde{V}$ on the according staggered grid, its boundary $\partial\tilde{V}$, and the normal with respect to the boundary $n_{\partial\tilde{V}}$. We have to express the terms regarding the staggered grid element $\tilde{V}$ by contributions from an element of the primal grid V (basenavierstokesstaggeredgrid.hh). Without loss of generality, we use the two-dimensional case as depicted in figure 3 together with the terms left, right, upper and lower. By primal grid, we mean the grid used for the mass balance equation which we have as a DUNE grid. As the momentum balance equation is shifted by half an element, we have to distinguish two cases: (a) The face normal is directed in the same direction as the velocity, the flux is within an element of the primal grid, cf. figure 3a. (b) The face normal is orthogonal to the velocity, the flux crosses the faces of the primal grid and it cannot be accounted for by a volume term, cf. figure 3b.

The storage term $\int_{\tilde{V}} \frac{\partial}{\partial t} \left(\varrho_\mathrm{g} v_\mathrm{g}\right)$ is split between the volume terms of two elements on the primal grid. The contribution of an element V on the primal grid to the residuum is $\frac{1}{2}|V|\frac{\partial}{\partial t} \left(\varrho_\mathrm{g} v_\mathrm{g,left}\right)$ for the velocity degree of freedom located on the left, cf. figure 3a.

For the inertia term $\int_{\partial\tilde{V}} \left(\varrho_\mathrm{g} v_\mathrm{g} v_\mathrm{g}^\intercal\right) \cdot n_{\partial\tilde{V}}$ we have the two cases. (a) v_g is co-directed to $n_{\partial\tilde{V}}$. The flux across the edge is a volume term $\Delta x_2 \varrho_\mathrm{g} v_\mathrm{g,avg} v_\mathrm{g,up}^\intercal \cdot n_{\partial\tilde{V}}$ in the primal grid for the left half. As the velocity is not located in the middle of V, one velocity is replaced by the averaged velocity, the other velocity is obtained an upwinding with respect to the averaged velocity. (b) v_g is normal to $n_{\partial\tilde{V}}$. The contributions are considered as skeleton terms. One primal grid volume is covered by two velocity control volumes halves, cf. figure 3b. In the two adjacent elements, four velocities in the considered direction are present, but we need only two for the contributions from the parts on the left. This results in $\frac{1}{2}\Delta x_1 \varrho_\mathrm{g} v_\mathrm{g,left} v_\mathrm{g,left}^\intercal \cdot n_{\partial\tilde{V}}$ with one averaged velocity; for the density and the other velocity we use upwinding.

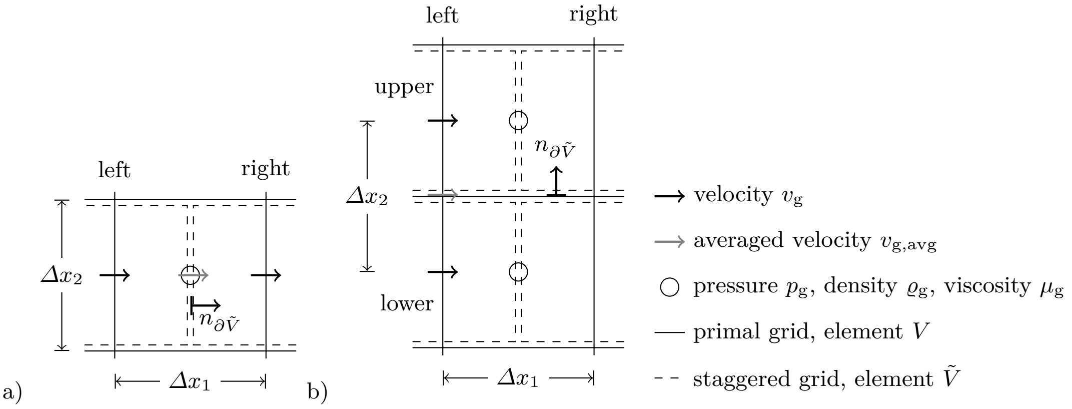


Fig. 3 Discretization with MAC scheme for momentum balance equation (a) velocity direct in same direction as the face normal (b) velocity normal to face normal

```
template<typename EG, typename LFSU, typename X, typename LFSV, typename R>
void alpha_volume(const EG& eg, const LFSU& lfsu, const X& x,
                  const LFSV& lfsv, R& r) const
{
    typedef typename LFSU::Traits::SizeType size_type;

    model_.localResidual().eval(eg.entity());

    int numVertices = x.size() / numEq;
    for (size_type dofIdx = 0; dofIdx < r.size(); ++dofIdx)
    {
        r.accumulate(lfsu, dofIdx,
                     model_.localResidual().residual
                     (dofIdx % numVertices)[dofIdx / numVertices]);
    }
}
```

Fig. 4 PDELab local operator to pass the residual from DuMu$^\mathrm{x}$

The viscous term $-\int_{\partial\tilde{V}} (\mu_\mathrm{g}\,\mathrm{grad}\,v_\mathrm{g})\cdot n_{\partial\tilde{V}}$ also requires the distinction of cases. (a) v_g is co-directed to $n_{\partial\tilde{V}}$. The gradient is the partial derivative in the direction of the velocity which is approximated by a finite difference. The contribution to the volume term is $-\Delta x_2 \mu_\mathrm{g} \frac{1}{\Delta x_1} (v_\mathrm{g,right} - v_\mathrm{g,left})$. (b) v_g is normal to $n_{\partial\tilde{V}}$. The contribution for the primal grid is considered as a skeleton term. The gradient is the partial derivative in the direction normal to the velocity. This leads to $-\frac{1}{2}\Delta x_1 \mu_\mathrm{g} \frac{v_\mathrm{g,upper} - v_\mathrm{g,lower}}{\Delta x_2}$ with an averaged viscosity.

The pressure term $\int_{\partial\tilde{V}} p_\mathrm{g} n_{\partial\tilde{V}}$ is considered as the volume term $\Delta x_2 p_\mathrm{g} n_{\partial\tilde{V}}$.

The other free-flow balance equations for mass, vapor mass fraction and energy are discretized with a cell-centered finite volume method. All occurring fluxes cross faces on the primal grid and are implemented as skeleton terms (basecomponentstaggeredgrid.hh, baseenergystaggeredgrid.hh).

4.2 Local operator wrapper

We need to include the residual and the Jacobian from the Darcy subdomain into the coupled residual and coupled Jacobian respectively. We use a local operator wrapper (multidomainlocaloperator.hh) that is a PDELab local operator passing the residual and Jacobian from DuMu$^\mathrm{x}$ to PDElab. The residual is passed element by element via alpha_volume, see the listing in figure 4.

After the local residual of the current entity is calculated by DuMu$^\mathrm{x}$, we accumulate it to the according local residual from PDELab. As it is not distinguishable and not relevant whether a contribution came from volume or skeleton terms, the whole residual is transferred via alpha_volume;

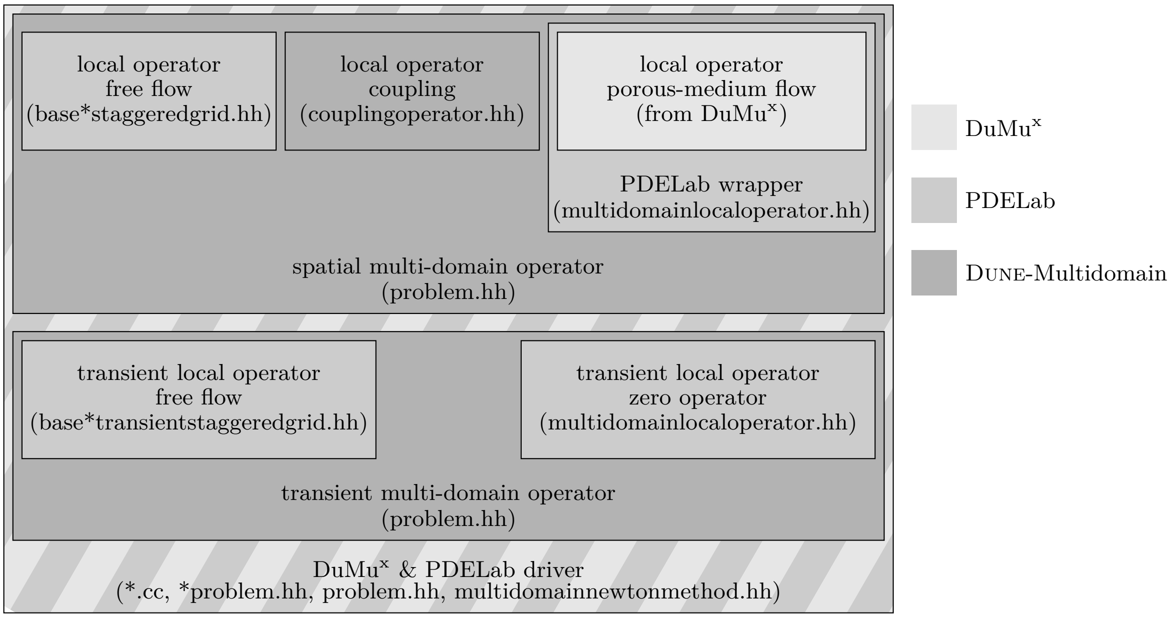


Fig. 5 Software structure of our monolithic coupled free-flow / porous-medium simulator for evaporation processes

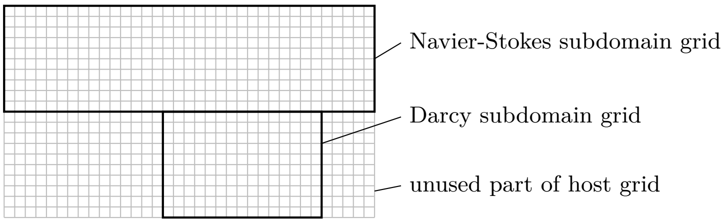


Fig. 6 Subdomains for Navier-Stokes and Darcy flow, unused grid elements

alpha_skeleton remains empty. The Jacobian is passed in a similar manner. Diagonal entries are passed via jacobian_volume and off-diagonal entries via jacobian_skeleton (multidomainlocaloperator.hh). The structure of the local operators is sketched in figure 5.

PDELab requires two local operators, a stationary and an instationary one. To compute the time integration with different methods, the instationary local operator returns the storage term. In DuMu^x the implicit Euler scheme is hard-coded and its contributions are already part of the Jacobian. We synchronize the time step between both parts and provide an empty instationary local operator for the Darcy subdomain to fulfill PDELab's interface requirements (multidomainlocaloperator.hh). Nevertheless, our concept is flexible enough to support extensions. In another project, we successfully added turbulence models like the k-ε model to the free flow.

4.3 Further adjustments

In this subsection, we present additionally necessary adjustments required for the coupling.

Our host grid is a YaspGrid in the tensor grid mode for graded meshes, which is finer towards the interface. DUNE-MultiddomainGrid provides and handles the subdomains. The upper and lower subdomains are for Navier-Stokes and Darcy respectively, see figure 6. The latter subdomain can be made narrower than the Navier-Stokes part to reduce the influence of the inflow and outflow boundary conditions which are sensitive for the free flow.

In general, DuMu^x is not prepared for subdomains from DUNE-MultiDomainGrid because the local operator acts on the multidomain grid view. As a result, we had to replace the base local operator and the assembler by a version that does not only check for neighboring elements, but additionally check whether neighboring elements are in the same subdomain (multidomaincclocalresidual.hh, multidomainccfvelementgeometry.hh).

DuMu$^\text{x}$ has a mechanism, called property system, to store and retrieve C++ types. All template arguments are wrapped in type tags, which provide the types. All classes get the type tag as a template argument instead of numerous template arguments. For the coupled problem, we have three type tags. The first for the Darcy subproblem, the second for the Navier-Stokes subproblem, and the third for the multidomain problem. The multidomain type tag offers access to the subproblem type tags and vice versa, acting as a go-between. The Darcy type tag inherits from the non-isothermal two-phase two-component Darcy problem from DuMu$^\text{x}$, which saves duplicate work. For the coupled problem, we need approximately 300 lines of codes to wrap all the types, required for function spaces, operators et cetera, in the property system (propertydefaults.hh).

DuMu$^\text{x}$ provides, as a major feature, implementations of physical properties and material laws. It uses a so called fluid state object that is constructed with the help of the primary variables. The object offers methods to access all resulting physical properties. The PDELab-based Navier-Stokes local operator does not suite well with this concept. We use a special fluid object which creates the fluid state for every secondary variable calculation (twocomponentfluid.hh). This reduces the setup code for the fluid state in the local operator. We waste a fair share for the recomputation of repeating fluid states. This can be reduced by a caching of fluid states or, even better, by an explicit Jacobian calculation.

The DuMu$^\text{x}$ Darcy model has an internal state that stores the variable switch, i. e., whether one phase vanished and subsequently a different primary variable must be used. To update its internal state and to check for variable switches, the Darcy model gets a copy of the solution vector after every Newton step. A variable switch alters the associated value in the solution vector. As a consequence, the solution vector has to be copied back (multidomainnewtonmethod.hh). The solution vector cannot be reused by passing a reference, because the degree of freedom ordering differs: DuMu$^\text{x}$ has one solution vector entry for every element, and the entry is a vector of all element unknowns. In contrast, PDELab uses a flat vector and calculates the according indices. Additionally, the solution vector created by DuMu$^\text{x}$ is too large, as it is not aware of the subdomains and it considers all elements. In a pre-simulation grid iteration, we collect all elements in the Darcy subdomain in a vector and use it for indirect accesses to the DuMu$^\text{x}$ solution vector. The function to copy the degrees of freedom is part of the problem and is passed as a lambda function to the Newton method. This allows to adjust the function to match the problem's needs (problem.hh).

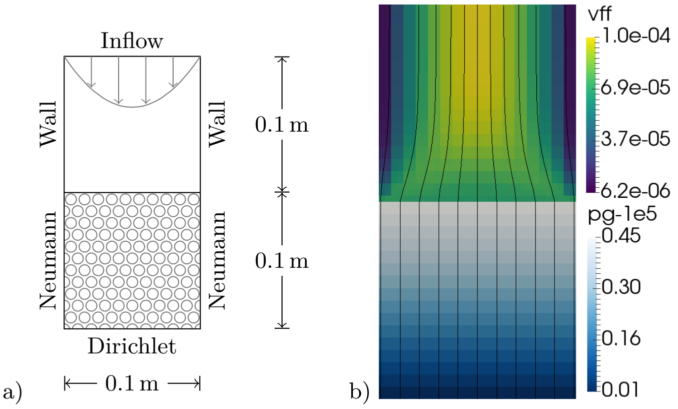
The goal is to reuse as much code as possible, but somewhere we need custom code to tie up the loose ends. This is the Newton method (multidomainnewtonmethod.hh), which is a mixture of DuMu$^\text{x}$ and PDELab code, glued together by some of the adjustments described above. Here, for instance, we keep the solution vectors in sync, trigger the evaluation of the local operators, call the time stepper, or let the results be written as VTK.

The implementation of a specific numeric simulation follows the DuMu$^\text{x}$ way. Most of the boiler plate code is part of abstract problems. Actual problems can alter properties and have to define boundary conditions, material properties and so on.

5 Numerical results

The following numerical tests demonstrate that our concept works and they showcase the effective operation of our implementation. We will (1) perform a basic numerical test, (2) simulate a lab experiment and (3) show a three-dimensional test. We will compare our results to the ones obtained with the box/box coupling scheme [14, 26] part of the current release of DuMu$^\text{x}$ 2.10.

In all tests we use the same physical properties from the lab experiment described by Mosthaf et al. [26]. If not stated otherwise, we use the following values, representing a two-dimensional system of dry air and a water filled box of granite sand. The fluid properties for water follow IAPWS [20], for air we assume an ideal gas with the constants from Reid et al. [28]. Properties of the porous medium are $\phi = 0.41$, $K = 2.65 \cdot 10^{-10}\,\mathrm{m}^2$, $\varrho_\mathrm{s} = 2700\,{}^\mathrm{kg}/_{\mathrm{m}^3}$, $\lambda_\mathrm{s} = 5.26\,{}^\mathrm{W}/_{\mathrm{m\,K}}$, $c_\mathrm{s} = 790\,{}^\mathrm{J}/_{\mathrm{kg\,K}}$, and $g = (0, \dots, -9.81\,{}^\mathrm{m}/_{\mathrm{s}^2})^\intercal$. With the Van Genuchten model we describe $k_{\mathrm{r},\alpha}$ and p_c using the constants $S_\mathrm{w,r} = 0.005$, $S_\mathrm{n,r} = 0.01$, $n_\mathrm{vG} = 8$ and $\alpha_\mathrm{vG} = 6.371 \cdot 10^{-4}\,{}^1/_\mathrm{Pa}$. For better convergence the functions from the Van Genuchten model are regularized with a linear approximation for saturations below



c)

Grid	Flux difference
8^2	$1.56 \cdot 10^{-10}$
16^2	$2.68 \cdot 10^{-10}$
32^2	$2.62 \cdot 10^{-10}$
64^2	$1.47 \cdot 10^{-10}$
128^2	$4.63 \cdot 10^{-10}$

Fig. 7 (a) Setup of mass conservation test (b) Steady-state solution after 300 s, on a grid with 256 uniform elements per subdomain, black: velocity stream lines, colors bottom: Darcy gas pressure $p_\mathrm{g}^\mathrm{pm} - 10^5\,\mathrm{Pa}$, colors top: velocity magnitude $\left|v^\mathrm{ff}\right|$ (c) Volume flux differences in $\mathrm{m^3/s}$ between influx volume at the top and outflux volume at the bottom for different grids with given number of elements per subdomain

1% and above 99%. For the Beavers-Joseph-Saffman condition we use $\alpha_\mathrm{BJ} = 1$. The initial values are for the free flow $p_\mathrm{g,init}^\mathrm{ff} = 10^5\,\mathrm{Pa}$, $T_\mathrm{init}^\mathrm{ff} = 298.15\,\mathrm{K}$ and the porous medium $p_\mathrm{g,init}^\mathrm{pm} = 10^5\,\mathrm{kg/m\,s^2}$, $T_\mathrm{init}^\mathrm{pm} = 298.15\,\mathrm{K}$ with both phases present.

Additionally, we have a simplified set of physical properties to reduce the computational costs. This involves a simpler fluid system for the free flow, including density and viscosity independent of pressure and temperature, and a pressure-independent binary diffusion coefficient. Further in the porous medium, the tortuosity $\tau = 0.6$ is constant instead of following the law of Millington and Quirk, and the thermal conductivity is calculated according to the model from Johanson instead of the one from Somerton.

5.1 Conservation of normal flow

The first test checks the conservation of mass for a simplified isothermal, single-phase setting, inspired by an example from Kanschat and Rivière [22]. It does not include evaporation and no water is in the system, the porous medium is completely dry $S_\mathrm{l} = 0$ and the air has no vapor mass fraction $X_\mathrm{g}^\mathrm{w,ff} = X_\mathrm{g}^\mathrm{w,pm} = 0$. The air flow enters the free flow subdomain with a parabolic shaped velocity profile $v_\mathrm{top}^\mathrm{ff} = (0, -0.04x_1(0.1 - x_1)\,\mathrm{m/s})^\intercal$ from the top, see figure 7a. The left and right boundaries are walls for the free-flow region and Neumann no-flow conditions for the porous medium. At the bottom of the porous medium a Dirichlet boundary condition fixes the pressure to $p_\mathrm{g,bottom} = 10^5\,\mathrm{Pa}$. To preserve the isothermal and single-phase flow nature of the test, Dirichlet boundary conditions $T = T_\mathrm{init}$ and $X_\mathrm{g}^\mathrm{w} = 0$ are imposed on all boundaries. The setup is chosen to measure the flux differences across the interface. The air entering the free flow at the top must cross the interface and leave the bottom of the porous medium, as the sides are leak-tight.

The steady-state simulation results shown in figure 7b accord with the ones from Kanschat and Rivière [22]. In the free flow the initial parabolic velocity profile evens towards the interface. Then the flow in the porous medium is almost parallel downwards. The free-flow pressure remains constant, in the porous medium a pressure gradient evolves. To calculate the volume flux loss we integrate over the velocities in the top most row of the free flow elements and compare it to the integrated Darcy velocities which are computed from the two bottom most rows of the porous medium elements. The differences are given in the table of figure 7c. They are near the accuracy of the linear solver and the finite difference used for the Jacobian, and are four orders of magnitude smaller compared to the volume flux of $\frac{2}{3} \cdot 10^{-5}\,\mathrm{m^3/s}$.

The result obtained from the box/box scheme is unexpected as the temperature in the free-flow region raises by 0.9 K and oscillations occur for the free-flow velocity near the corners above the coupling interface.

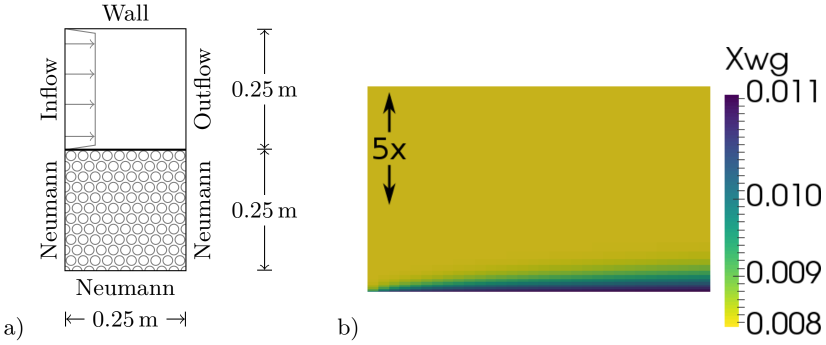


Fig. 8 Wind tunnel (a) Geometry and boundary conditions (b) Cut-out of the free flow subdomain above the interface $x_1 \in \{0.25\,\mathrm{m}, 0.28\,\mathrm{m}\}$ from figure 9d, five times vertically exaggerated

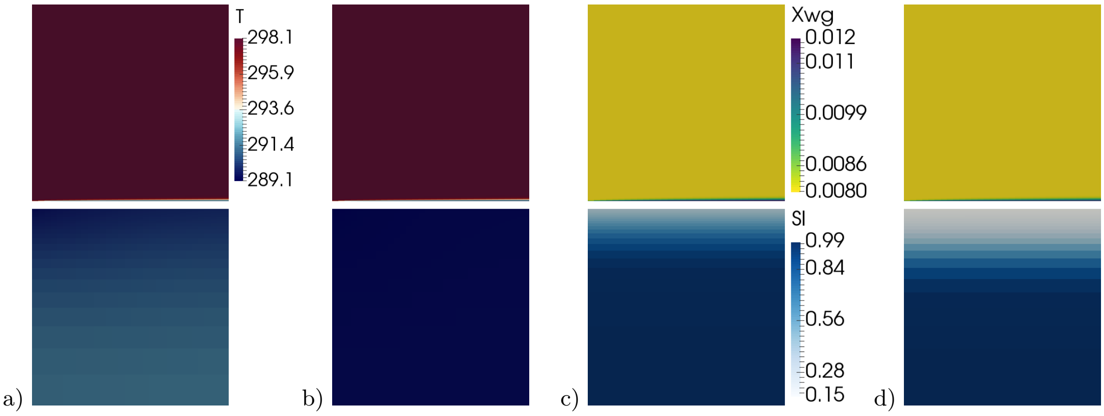


Fig. 9 Wind tunnel (a) Temperature T at $t = 1\,\mathrm{d}$ (b) $t = 3\,\mathrm{d}$ (c) Saturation of the liquid phase in the porous medium S_l and vapor mass fraction in free flow X_g^w at $t = 1\,\mathrm{d}$ (d) $t = 3\,\mathrm{d}$

5.2 Wind tunnel: Tangential flow with evaporation

We simulate the lab experiment by Mosthaf et al. [26] and compare our results with the ones obtained by the box-scheme-based implementation used in their paper and the ones with the simplified physics. The setup is a tangential flow over an initially saturated, sand-filled box. The box is insulated and located on a balance, to measure the evaporated water mass. The computational domain $[0, 0.25\,\mathrm{m}] \times [0, 0.5\,\mathrm{m}]$ is divided at $x_1 = 0.25\,\mathrm{m}$ into the porous medium in the lower and the free flow domain in the upper half, see figure 8a. For the porous medium all boundary conditions are Neumann no-flow, representing the insulated box. For the free flow the boundary conditions are an inflow boundary condition on the left, a wall at the top and outflow on the right. The initial time step is one millisecond, the maximal time step is one hour; the simulation lasts six days. The grid is graded towards the interface, that means an element is higher by a constant factor than the neighboring element closer to the interface. The elements next to the interface are $2 \cdot 10^{-4}\,\mathrm{m}$ high. The grid has 1024 elements per subdomain.
In figure 9 the effects of the evaporation are visible, the water saturation next to the interface decreases. The porous medium is cooled by the evaporation and the box cools down. The free flow area with a higher water mass fraction and lower temperature is thin, figure 8b shows a five times exaggerated cut-out.

The evaporation rate is an important result as it describes the loss of soil water. We recalculated the evaporation rates from Mosthaf et al. [26] with the current version of DuMu$^\mathrm{x}$ and superimpose the graphs in figure 10a. The box/box scheme leads to slightly higher evaporation rates. The difference can be explained with the different boundary conditions. The box/box scheme enforces the same water mass fraction across the interface, while for the MAC/FVM the water mass exchange to the free flow is driven by its gradient. The simplified physics does hardly alter the evaporation rate

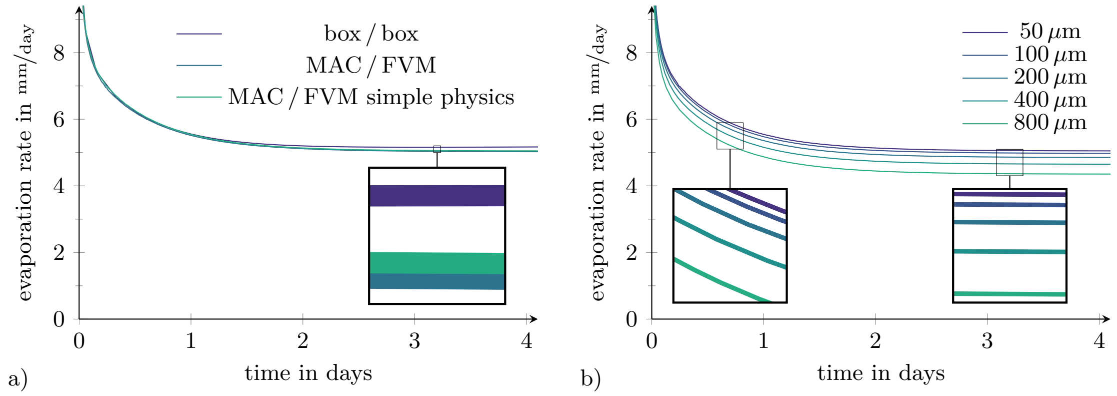


Fig. 10 Wind tunnel evaporation rates (a) Comparing MAC/FVM with box/box scheme, thickness of thinnest element is 50 μm (b) Grid convergence with same number of grid elements but a higher grading factor, the thickness of the thinnest element is indicated, MAC/FVM scheme with simple physics

The exchange across the interface drives the processes. We examine the effect of smaller elements near the interface. Figure 10b shows a series of evaporation rates from simulations with finer grids near the interface. The grids have a constant number of elements, the grading factor is modified within the series. The evaporation rates increase for smaller elements near the interface, but the increase becomes smaller, indicating convergence.

The box scheme has difficulties with the degrees of freedom located at the end points of the coupling interface. For the MAC/FVM scheme, we do not observe such difficulties. The run-times are comparable, with an advantage of the assembly and a better Newton convergence for the box/box scheme and faster solving of the linear system for the MAC/FVM scheme. In the course of implementing the MAC/FVM scheme, we could gain a speed-up of an order of magnitude compared to the code used in the cited sources, but both methods profit from the speedup. The main reason for the speedup is a caching of the fluid system and rearranging of formulas to reduce the evaluation of expensive mathematical operators.
The simplified physics reduces the run-times by a factor of 1.3 to 2.3.

5.3 Three-dimensional wind tunnel

In the last test case we extend the wind tunnel to three dimensions. We scale the domain size down and relatively enlarge the free flow subdomain, to allow lateral diffusion, see figure 11. The front is a wall, the back is a symmetry boundary condition to simulate only half the wind tunnel. The inflow velocity is reduced to $v^{\mathrm{ff}} = (0.1\,\mathrm{m}/\mathrm{s}, 0, 0)^{\mathsf{T}}$ to increase diffusive effects. The grid has 768 elements in the porous medium and 6 240 in the free flow subdomain, see figure 12. This leads to a system with 40 816 degrees of freedom. The initial time step size is five seconds, the maximal time step is ten minutes and the simulation spans one hour.
The results are similar to the two-dimensional case, see figure 12. The increased diffusion spreads the cooler air further away from the interface. The coupling conditions and the implementation work well in three dimensions.
There are no published results with a three-dimensional setup for comparison. The implementation using the box/box scheme does only work for two-dimensional setups.

6 Conclusions

In this paper we introduce a coupling concept and an according implementation for a single-phase, two-component, non-isothermal Navier-Stokes flow and a two-phase, two-component, non-isothermal Darcy flow. It consists of a FVM and a MAC scheme, conserves mass, and is more robust than the

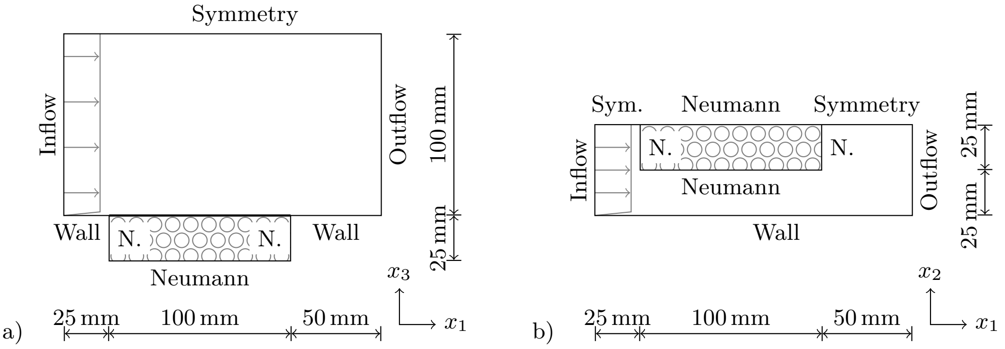


Fig. 11 Setup of three-dimensional wind tunnel example (a) side view (b) top view

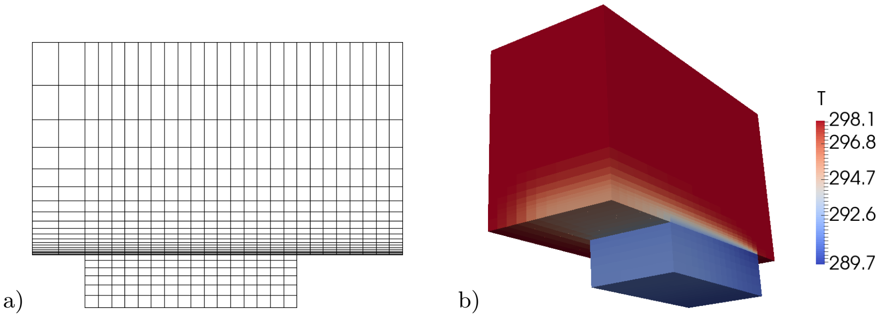


Fig. 12 Three-dimensional wind tunnel (a) side view of grid (b) temperature viewed from behind at $t = 1\,\mathrm{h}$

box/box scheme. We showcase our implementation which is fast enough to simulation lab-scale evaporation experiment and which provides features necessary for applications. Simulations of different application based on this code are described in Grüninger [16].

If we started our project today, we would try to avoid the complexity of two different discretization modules. Instead, we would aim for an implementation purely in DuMu^x. Alternatively, the degree of freedom handling offered by dune-functions [13] looks appealing to our problem. Probably the obstacles of coupling DuMu^x and PDELab would be lower, if both modules were based on dune-functions.

On top of this work further effects could be included like solar radiation or vegetation. With finer grids and smaller time steps turbulent effects could be simulated, but the assembly time and the memory restrictions by the direct linear solvers limit such applications. Interested users can implement turbulence models similar to Fetzer et al. [14] and use DUNE's parallel capacities.
Using our implementation the coupling conditions can be modified and their effects examined. Also decoupling strategies are worth investigating. On top of our work, the iterative coupling of the two physical subdomains from Discacciati and Quarteroni [12] could be applied. Alternatively the decoupling from Masson et al. [24], which separates the Navier-Stokes equation from all other equations in an iterative decoupling, could be beneficial.

How to reproduce the numerical results The source code is available from DuMu^x-pub under GPL version 2 or, at your option, any later version, `https://git.iws.uni-stuttgart.de/dumux-pub/Grueninger2017b.git`. It provides the script installGrueninger2017b.sh that downloads and compiles all required DUNE and DuMu^x modules. The README contains further information concerning the executables and how to reproduce the numerical results of this work.

Acknowledgments The authors C. G., B. F. and R. H. would like to thank the German Research Foundation (DFG) for financial support of the project within the Cluster of Excellence in Simulation Technology (EXC 310/2) at the University of Stuttgart. Some of the authors are members of the International Research Training Group NUPUS funded by the German Research Foundation (DFG) and Netherlands Organization for Scientific Research (NWO), we thank for their support.

References

1. K. Baber, K. Mosthaf, B. Flemisch, R. Helmig, S. Müthing, and B. Wohlmuth. Numerical scheme for coupling two-phase compositional porous-media flow and one-phase compositional free flow. *IMA Journal of Applied Mathematics*, 77(6):887–909, 2012. doi: 10.1093/imamat/hxs048.
2. P. Bastian, M. Blatt, A. Dedner, C. Engwer, R. Klöfkorn, R. Kornhuber, M. Ohlberger, and O. Sander. A generic grid interface for parallel and adaptive scientific computing. Part II: Implementation and tests in DUNE. *Computing*, 82(2–3):121–138, 2008. doi: 10.1007/s00607-008-0004-9.
3. P. Bastian, M. Blatt, A. Dedner, C. Engwer, R. Klöfkorn, M. Ohlberger, and O. Sander. A generic grid interface for parallel and adaptive scientific computing. Part I: Abstract framework. *Computing*, 82(2–3):103–119, 2008. doi: 10.1007/s00607-008-0003-x.
4. P. Bastian, F. Heimann, and S. Marnach. Generic implementation of finite element methods in the Distributed and Unified Numerics Environment (DUNE). *Kybernetika*, 46(2):294–315, 2010. URL http://hdl.handle.net/10338.dmlcz/140745.
5. M. Blatt, A. Burchardt, A. Dedner, C. Engwer, J. Fahlke, B. Flemisch, C. Gersbacher, C. Gräser, F. Gruber, C. Grüninger, D. Kempf, R. Klöfkorn, T. Malkmus, S. Müthing, M. Nolte, M. Piatkowski, and O. Sander. The Distributed and Unified Numerics Environment, Version 2.4. *Archive of Numerical Software*, 100(4):13–29, May 2016. doi: 10.11588/ans.2016.100.26526.
6. H. Class, R. Helmig, and P. Bastian. Numerical simulation of non-isothermal multiphase multicomponent processes in porous media. 1. an efficient solution technique. *Advances in Water Resources*, 25(5):533–550, 2002. doi: 10.1016/S0309-1708(02)00014-3.
7. W. Dahmen, T. Gotzen, S. Müller, and M. Rom. Numerical simulation of transpiration cooling through porous material. *International Journal for Numerical Methods in Fluids*, 76(6):331–365, 2014. doi: 10.1002/fld.3935.
8. D. Das, V. Nassehi, and R. Wakeman. A finite volume model for the hydrodynamics of combined free and porous flow in sub-surface regions. *Advances in Environmental Research*, 7(1):35–58, 2002. doi: 10.1016/S1093-0191(01)00108-3.
9. T. A. Davis. A column pre-ordering strategy for the unsymmetric-pattern multifrontal method. *ACM Trans. Math. Softw.*, 30(2):165–195, 2004. doi: 10.1145/992200.992205.
10. T. Defraeye. Advanced computational modelling for drying processes – a review. *Applied Energy*, 131(C): 323–344, 2014. doi: 10.1016/j.apenergy.2014.06.027.
11. T. Defraeye, B. Blocken, and J. Carmeliet. Analysis of convective heat and mass transfer coefficients for convective drying of a porous flat plate by conjugate modelling. *International Journal of Heat and Mass Transfer*, 55(1-3):112–124, 2012. doi: 10.1016/j.ijheatmasstransfer.2011.08.047.
12. M. Discacciati and A. Quarteroni. Navier-Stokes/Darcy coupling: modeling, analysis, and numerical approximation. *Revista Matemática Complutense*, 22(2):315–426, 2009. doi: 10.5209/rev_REMA.2009.v22.n2.16263.
13. C. Engwer, C. Gräser, S. Müthing, and O. Sander. The interface for functions in the dune-functions module. *CoRR*, abs/1512.06136, 2015. URL http://arxiv.org/abs/1512.06136.
14. T. Fetzer, K. M. Smits, and R. Helmig. Effect of turbulence and roughness on coupled porous-medium/free-flow exchange processes. *Transport in Porous Media*, pages 1–30, 2016. doi: 10.1007/s11242-016-0654-6.
15. B. Flemisch, M. Darcis, K. Erbertseder, B. Faigle, A. Lauser, K. Mosthaf, S. Müthing, P. Nuske, A. Tatomir, M. Wolff, and R. Helmig. DuMu^x: DUNE for multi-{phase, component, scale, physics, . . . } flow and transport in porous media. *Advances in Water Resources*, 34(9):1102–1112, 2011. doi: 10.1016/j.advwatres.2011.03.007.
16. C. Grüninger. *Numerical Coupling of Navier-Stokes and Darcy Flow for Soil-Water Evaporation.* PhD thesis, University of Stuttgart, May 2017.
17. F. H. Harlow and J. E. Welch. Numerical calculation of time-dependent viscous incompressible flow of fluid with free surface. *Physics of Fluids*, 8(12):2182–2189, 1965. doi: 10.1063/1.1761178.
18. R. Helmig. *Multiphase flow and transport processes in the subsurface: a contribution to the modeling of hydrosystems.* Environmental engineering. Springer, 1997. ISBN 9783540627036.
19. J. Hommel, S. Ackermann, M. Beck, B. Becker, H. Class, T. Fetzer, B. Flemisch, D. Gläser, C. Grüninger, K. Heck, A. Kissinger, T. Koch, M. Schneider, G. Seitz, and K. Weishaupt. Dumux 2.10.0, Sept. 2016.
20. IAPWS. Revised release on the IAPWS industrial formulation 1997 for the thermodynamic properties of water and steam, 2007. URL http://www.iapws.org/relguide/IF97-Rev.pdf.
21. V. A. Jambhekar, R. Helmig, N. Schröder, and N. Shokri. Free-flow–porous-media coupling for evaporation-driven transport and precipitation of salt in soil. *Transport in Porous Media*, 110(2):251–280, 2015. doi: 10.1007/s11242-015-0516-7.
22. G. Kanschat and B. Rivière. A strongly conservative finite element method for the coupling of Stokes and Darcy flow. *Journal of Computational Physics*, 229(17):5933–5943, 2010. doi: 10.1016/j.jcp.2010.04.021.
23. W. J. Layton, F. Schieweck, and I. Yotov. Coupling fluid flow with porous media flow. *SIAM Journal on Numerical Analysis*, 40(6):2195–2218, 2002. URL http://www.jstor.org/stable/4100990.
24. R. Masson, L. Trenty, and Y. Zhang. Coupling compositional liquid gas darcy and free gas flows at porous and free-flow domains interface. *Journal of Computational Physics*, 321:708–728, 2016. doi: 10.1016/j.jcp.2016.06.003.

25. K. Mosthaf, K. Baber, B. Flemisch, R. Helmig, A. Leijnse, I. Rybak, and B. Wohlmuth. A coupling concept for two-phase compositional porous-medium and single-phase compositional free flow. *Water Resources Research*, W10522:19 PP, 2011. doi: 10.1029/2011WR010685.
26. K. Mosthaf, R. Helmig, and D. Or. Modeling and analysis of evaporation processes from porous media on the REV scale. *Water Resources Research*, 50, 2014. doi: 10.1002/2013WR014442.
27. S. Müthing. *A Flexible Framework for Multi Physics and Multi Domain PDE Simulations*. PhD thesis, University of Stuttgart, January 2015.
28. R. Reid, J. Prausnitz, and B. Poling. *The properties of gases and liquids*. McGraw Hill Book Co.,New York, NY, Jan 1987.
29. D. Rind, C. Rosenzweig, and M. Stieglitz. The role of moisture transport between ground and atmosphere in global change. *Annual Review of Energy and the Environment*, 22(1):47–74, 1997. doi: 10.1146/annurev.energy.22.1.47.
30. I. Rybak, J. Magiera, R. Helmig, and C. Rohde. Multirate time integration for coupled saturated/unsaturated porous medium and free flow systems. *Computational Geosciences*, 19(2):299–309, 2015. doi: 10.1007/s10596-015-9469-8.
31. P. G. Saffman. On the boundary condition at the surface of a porous medium. *Studies in Applied Mathematics*, 50(2):93–101, 1971. doi: 10.1002/sapm197150293.
32. R. Usha, A. Anjalaiah, and Y. V. S. S. Sanyasiraju. Dynamics of a pre-lens tear film after a blink: Model, evolution, and rupture. *Physics of Fluids*, 25(11), 2013. doi: 10.1063/1.4831795.
33. J. Vanderborght, T. Fetzer, K. Mosthaf, K. M. Smits, and R. Helmig. Heat and water transport in soils and across the soil-atmosphere interface: 1. Theory and different model concepts. *Water Resources Research*, 2017. doi: 10.1002/2016WR019982.