Flux Postprocessing and Semi-Analytical Particle Tracking for Finite-Element-Type Models of Variably Saturated Flow in Porous Media

Dissertation

der Mathematisch-Naturwissenschaftlichen Fakultät

der Eberhard Karls Universität Tübingen

zur Erlangung des Grades eines

Doktors der Naturwissenschaften

(Dr. rer. nat.)

vorgelegt von

Philipp Selzer

aus

Hamburg

Tübingen

2021

Gedruckt mit Genehmigung der Mathematisch-Naturwissenschaftlichen Fakultät der Eberhard Karls Universität Tübingen.

Tag der mündlichen Qualifikation:	30.05.2022
Dekan:	Prof. Dr. Thilo Stehle
1. Berichterstatter:	Prof. Dr.-Ing. Olaf A. Cirpka
2. Berichterstatter:	Prof. René Therrien, Ph.D.
3. Berichterstatter	Prof. Dr.-Ing. Rainer Helmig

Declaration of contributions

The present dissertation is largely based on two publications, which are named below.

Selzer, P., Cirpka, O.A., 2020. Postprocessing of standard finite element velocity fields for accurate particle tracking applied to groundwater flow. Computational Geosciences. 24 (4), 1605–1624. https://doi.org/10.1007/s10596-020-09969-y.

Contributions in Selzer and Cirpka (2020):

Philipp Selzer: Conceptualization, Methodology, Software implementation, Validation, Writing – original draft. Refinement and adaptation of the $\mathcal{RTN}_0$-projection, Idea of the FVM on simplices, Idea and derivation of the analytical solutions for the particle trajectories in divergent flow fields on simplices, Development of an optimal strategy for splitting grids of conforming cubes into grids of conforming tetrahedra such that the grid effects of the lowest-order FVM are minimized (applied but not described in Selzer and Cirpka (2020)), Implementation of the methods
Olaf A. Cirpka: Conceptualization, Methodology, Writing – review & editing, Supervision, Funding acquisition. Idea of an $\mathcal{RTN}_0$-projection, Idea and derivation of an analytical solution for groundwater flow on a rectangle with in- and outflow windows including anisotropy as a ratio

Selzer, P., Allgeier, J., Therrien, R., and Cirpka, O.A., 2021. Finite-volume flux reconstruction and semi-analytical particle tracking on triangular prisms for finite-element-type models of variably-saturated flow. Advances in Water Resources, 154:103944. https://doi.org/10.1016/j.advwatres.2021.103944.

Contributions in Selzer et al. (2021):

Philipp Selzer: Conceptualization, Methodology, Software implementation, Writing – original draft. Idea of the finite-volume flux reconstruction, Writing of the original draft besides the chapter on the sub-catchment-scale application
Jonas Allgeier: Validation of the software, Writing - original draft. Writing of the chapter on the sub-catchment-scale application
René Therrien: Writing - review & editing, Supervision
Olaf A. Cirpka: Conceptualization, Methodology, Writing – review & editing, Supervision, Funding acquisition. Idea of the mixed coordinate system

Abstract

Aquifers are predominantly conceptualized as porous media. In real-world applications, the material properties and boundary conditions are heterogeneous requiring numerical models for an accurate description of flow and transport processes. Such numerical models yield discretized hydraulic heads and velocities on nodes or within elements. Particle tracking is a computationally advantageous and fast scheme to determine trajectories and travel times. Accurate particle tracking relies on conforming velocity fields that ensure local mass conservation in elements, and a continuous normal velocity component on element boundaries. While cell-centered finite-volume and mixed finite-element methods result in conforming velocity fields by definition, this is not the case for continuous Galerkin methods, such as the standard finite element method (FEM), and some finite-difference discretizations. Nonetheless standard FEM and also finite difference methods (FDM), formulated in finite-element terms, are often used for subsurface flow modeling because they yield a continuous approximations of hydraulic heads, and easily handle unstructured grids and material anisotropy.

Acknowledging these advantages and the wide-spread use of finite-element-type simulations, the aim of this thesis is to present a novel framework for computing conforming velocity fields and accurate particle trajectories for finite-element type primal solutions of variably saturated flow in porous media. In this thesis, two different postprocessing methods to compute conforming velocity fields based on non-conforming primal solutions and semi-analytical particle tracking techniques for triangles, tetrahedra, and triangular prisms are presented.

The first postprocessing method is a projection mapping a non-conforming, element-wise given velocity field onto a conforming velocity field in lowest-order Raviart-Thomas-Nédélec ($\mathcal{RTN}_0$) space, which meets the requirements of accurate particle tracking. The projection is based on minimizing the difference in the hydraulic gradients at the element centroids between a finite-element-type primal solution and the hydraulic gradients consistent with the $\mathcal{RTN}_0$ velocity field imposing element-wise mass conservation for variably saturated flow in porous media. The results of the $\mathcal{RTN}_0$-projection are close to those of a cell-centered finite volume method (FVM) defined for comparison and the finite-element-type primal solution. Consistency and convergence of the $\mathcal{RTN}_0$-projection are empirically shown for saturated flow based on a test case including hydraulic anisotropy. However, the $\mathcal{RTN}_0$-projection requires solving a large indefinite system of linear equations, which strongly reduces the choice of appropriate solvers, and occasionally shows numerical artifacts like velocities of wrong magnitude and unphysical direction in subregions of the domain in special cases.

The second postprocessing technique reconstructs a cell-centered finite-volume solution from a finite-element-type primal solution of variably saturated flow in porous media to obtain conforming, mass-conservative fluxes in $\mathcal{RTN}_0$-space. The method is exemplified for triangular prisms because this is one of the most common elements used for catchment-scale subsurface discretization. The finite-volume flux reconstruction only solves a linear elliptic problem whose size is on the order of the number of elements, which is computationally much faster than

solving the $\mathcal{RTN}_0$-projection or the initial Richards equation describing non-linear transient and variably saturated flow. Compared to other postprocessing schemes, the finite-volume flux reconstruction is numerically stable, fast to compute, and does not induce severe numerical artifacts when applied to heterogeneous domains with strongly varying velocities. Its advantage lies in the application to non-linear flow laws and transient problems, because it is assumed that the non-linearities are already solved by the primal solution and the transient term can be treated as a change in storage. It is shown that the results of the finite-volume flux reconstruction are close to the finite-element-type primal solution for variably saturated three-dimensional flow with heterogeneous material properties and boundary conditions.

Semi-analytical particle tracking is based on element-wise analytical solutions for particle trajectories and associated travel times given a numerically approximated velocity field. This facilitates the direct computation of the spatial coordinates where the particle exits an element from its entry point and the attributes of the element-wise velocity field. In this thesis, element-wise analytical solutions are given for triangles, tetrahedra, and triangular prisms using the linear average velocity field derived from the fluid fluxes in $\mathcal{RTN}_0$-space.

The contribution of this thesis is to provide a computational framework for approximating conforming, mass-conservative velocity fields and, based on this, semi-analytical particle tracking routines applicable to finite-element-type models of variably saturated flow in porous media on unstructured grids.

Kurzfassung

Grundwasserleiter werden zumeist als poröse Medien konzeptualisiert. In der Realität sind deren Materialkonstanten und Randbedingungen variabel. Dies verlangt numerische Modelle, um Strömungs- und Transportprozesse zuverlässig zu simulieren. Diese numerischen Modelle ergeben diskrete hydraulische Potentiale und Geschwindigkeiten auf Knoten oder in Elementen. Particle Tracking hat sich als genaue und effektive Methode etabliert, um Bahnlinien und Verweilzeiten zu bestimmen. Ein stabiles Particle Tracking erfordert konforme Geschwindgkeitsfelder, die in den Elementen lokal masseerhaltend sind und eine kontinuierliche Normalkomponente des Geschwindigkeitsvektors auf Elementgrenzen aufweisen. Während zell-zentrierte finite Volumen und gemischte Finite-Elemente-Verfahren per Definition konforme Geschwindigkeitsfelder ergeben, ist dies für kontinuierliche Galerkin-Methoden, wie der Standard-Finite-Elemente-Methode (FEM), und einige Finite-Differenzen-Diskretisierungen, nicht der Fall. Standard FEM und verwandte Methoden, wie Finite-Differenzen-Methoden formuliert in Finite-Elemente-Begriffen, werden dennoch häufig für die Strömungsmodellierung im Untergrund verwendet, da sie eine kontinuierliche Approximation der Potentiale ergeben sowie auf einfache und natürliche Weise unstrukturierte Gitter und Materialanisotropie behandeln können.

Unter Anerkennung dieser Vorteile und der weiten Verbreitung von Simulationen aus der Familie der finiten Elemente ist es das Ziel dieser Doktorarbeit, neuartige Methoden vorzustellen, welche die Berechnung konformer Geschwindigkeitsfelder und exakter Partikeltrajektorien für Modelle aus der Familie der finiten Elemente zur Berechnung variabel gesättigter Strömungen in porösen Medien ermöglichen. In dieser Doktorarbeit werden zwei verschiedene Postprocessing-Methoden zur Berechnung konformer Geschwindigkeitsfelder basierend auf nicht-konformen Primärlösungen sowie semi-analytische Particle-Tracking-Methoden für Dreiecke, Tetraeder und Dreicksprismen vorgestellt.

Die erste Postprocessing-Methode beruht auf einer Projektion, welche ein nicht-konformes, elementweise gegebenes Geschwindgkeitsfeld auf ein konformes Geschwindigkeitsfeld im Raviart-Thomas-Nédélec-Raum geringster Ordnung ($\mathcal{RTN}_0$) abbildet, das die Voraussetzungen für ein exaktes Particle Tracking erfüllt. Die Projektion basiert auf einer Minimierung der Differenz zwischen den hydraulischen Gradienten einer Standard-Galerkin-Finite-Elemente-Lösung und den Gradienten gemäß einer Lösung im $\mathcal{RTN}_0$-Raum geringster Ordnung in den Elementschwerpunkten unter der Nebenbedingung lokaler elementweiser Massenerhaltung für variabel gesättigte Strömungen in porösen Medien. Die Ergebnisse der $\mathcal{RTN}_0$-Projektion für gesättigte Strömungen sind ähnlich derer einer zell-zentrierten Finite-Volumen-Methode (FVM). Konsistenz und Konvergenz der $\mathcal{RTN}_0$-Projektion werden für gesättigte Strömungen anhand eines Testfalls unter Berücksichtigung hydraulischer Anisotropie empirisch gezeigt. Jedoch erfordert das Postprocessing die Lösung großer nicht-definiter Gleichungssysteme, was die Auswahl geeigneter Löser stark einschränkt. Außerdem führt das Verfahren in Spezialfällen zu numerischen Artefakten wie Geschwindgkeiten, die in Untermodellgebieten im Betrag falsch und in der Richtung unphysikalisch sind.

Die zweite Postprocessing-Methode rekonstruiert eine zell-zentrierte Finite-Volumen-Lösung auf Basis einer Primärlösung aus der Familie der finiten Elemente, um konforme, massenerhaltende Flüsse im $\mathcal{RTN}_0$-Raum zu erhalten, und wird auf variabel gesättigte Strömungen in porösen Medien angewandt. Die Methode wird beispielhaft für Dreicksprismen beschrieben, da diese für die Diskretisierung des Untergrunds auf der Einzugsgebietsskala häufig verwendet werden. Die Finite-Volumen-Flussrekonstruktion löst lediglich ein elliptisches Problem von der Ordnung der Anzahl der Elemente, was sehr viel schneller zu berechnen ist als die $\mathcal{RTN}_0$-Projektion oder die zu Grunde liegende Richards Gleichung für nicht-stationäre, variabel-gesättigte Strömungen. Verglichen mit anderen Postprocessing-Verfahren ist die Finite-Volumen-Flussrekonstruktion numerisch stabil, schnell zu berechnen und induziert keine wesentlichen numerischen Artefakte, wenn sie auf heterogene Gebiete mit stark variierenden Geschwindigkeiten angewandt wird. Ihr Vorteil liegt in der Anwendung auf nicht-lineare Strömungsgleichungen und nicht-stationäre Probleme, da angenommen wird, dass die Nichtlinearitäten bereits in der Primärlösung gelöst wurden und der nicht-stationäre Term als Quell-/Senkenterm behandelt werden kann. Anhand von Beispielen wird gezeigt, dass die Ergebnisse der Finite-Volumen-Flussrekonstruktion nahe der Primärlösungen aus der Familie der finiten Elemente unter Berücksichtigung heterogener Materialeigenschaften und Randbedingungen sind.

Das semi-analytische Particle Tracking basiert auf elementweisen analytischen Lösungen für die Partikeltrajektorien und assoziierten Verweilzeiten auf der Grundlage eines numerisch approximierten Geschwindigkeitsfeldes. Dies ermöglicht es, aus dem Eintrittspunkt eines Teilchens und dem numerisch approximierten Geschwindigkeitsfeld innerhalb eines Elementes direkt die Austrittskoordinaten des Teilchens zu berechnen. In dieser Doktorarbeit werden elementweise analytische Lösungen für Dreiecke, Tetraeder und Dreiecksprismen auf Basis der Abstandsgeschwindigkeiten gegeben, welche von den Flüssen im $\mathcal{RTN}_0$-Raum abgeleitet werden.

Diese Doktorarbeit stellt ein rechnergestütztes Methodengerüst zur Verfügung, um konforme und elementweise massenerhaltende Geschwindigkeitsfelder auf der Grundlage von Strömungslösungen aus der Familie der finiten Elemente zu approximieren. Hierauf basierend werden semi-analytische Particle-Tracking-Algorithmen entwickelt, die auf variabel gesättigte Strömungen in porösen Medien angewandt werden können.

Acknowledgment

I would like to thank:

- my main supervisor Olaf A. Cirpka who not only shared his ideas, which are part of this work, and enthusiasm with me, but also supported me constantly during the time of my PhD, gave me the freedom to develop own ideas, and always had time for a meeting despite his busy schedule. Olaf, I appreciate it very much that I had the opportunity to work with you – *Vielen Dank.*
- René Therrien for co-supervising my PhD-project and for explaining me the internal mechanisms and computational methods of HydroGeoSphere – *Merci beaucoup en particulier pour ton accueil très sympathique à Québec.*
- Claus P. Haslauer for helping me with my first scientific steps within the topic which ultimately lead to my PhD – *Vielen Dank.*
- Rainer Helmig for being my third *Berichterstatter*, reading my thesis, and writing an expertise.
- Jonas Allgeier for testing my codes intensively and being patient with facing bugs on the way – *Vielen Dank.*
- Olaf Ippisch from the Technical University of Clausthal for a very insightful discussion on the numerics of Richards' equation and Ole Klein from the Interdisciplinary Center for Scientific Computing at Heidelberg University for an other very insightful discussion on finite element methods and the numerics behind a possible $\mathcal{RTN}_0$-projection.
- Max Allmendinger for proofreading the mathematical notation of my first paper and many joyful discussions on mathematical topics.
- the German Research Foundation (DFG) for the funding I received via the Research Training Group (RTG) 1829 "Integrated Hydrosystem Modelling" and the Collaborative Research Center (CRC) 1253 "CAMPOS – Catchments as Reactors", and the Project DEAL for the open access funding for my first paper.
- all members and associates of the RTG – thanks to all people who made this scientific environment possible, especially my fellow PhD-students and Monika Jekelius.
- my family, especially my parents, who among others supported my long studies, my sister, and my grandparents – *Vielen Dank.*
- all my friends who joined me on my way and supported me in many situations – *Vielen Dank.*
- Elena for her love and support – Спасибо за всё. Я очень рад, что мы встретились.

Table of Contents

List of Figures

List of Tables

Notation

In general, throughout this thesis scalars and scalar-valued functions are denoted by standard letters, vectors and vector-valued functions are typeset in boldface lower-case letters, and matrices are denoted by boldface upper-case letters. However, every variable is introduced in the text, and seldom exceptions from the general rule are meant for better understanding and are clearly identifiable as such.

Operators

$\partial f/\partial x$	partial derivative of a function f with respect to x
$\partial(\cdot)$	boundary of a set
$\nabla(\cdot)$	gradient
$\nabla\cdot(\cdot)$	divergence
df/dx	total derivative of a function f with respect to x
$\|\cdot\|_2$	Euclidean norm
$\langle\cdot,\cdot\rangle$	standard inner product
$[\cdot]_F$	jump of a function on face F
$\|\cdot\|$	measure or cardinality of a set

Physical dimensions

M	Mass
L	Length
T	Time

Physical variables

c	concentration [ML^{-3}]
$\mathbf{D}$	dispersion tensor [$\mathrm{L}^2\mathrm{T}^{-1}$]
f	volumetric source/sink term per unit volume [T^{-1}]
h	hydraulic head [L]
k	direction-independent scalar hydraulic conductivity [LT^{-1}]
$\mathbf{K}$	hydraulic conductivity tensor [LT^{-1}]
k_r	relative permeability [−]
N	van Genuchten N [−]
n_e	effective porosity [−]
$\mathbf{q}$	specific discharge or Darcy velocity [LT^{-1}]

S_0	specific-storage coefficient in groundwater flow [L^{-1}]
S_e	effective saturation [$-$]
S_s	specific storage coefficient for variably saturated flow [L^{-1}]
$\mathbf{v}$	linear average velocity [LT^{-1}]
z	elevation head [L]
α	van Genuchten alpha [L^{-1}]
ψ	pressure head [L]
ρ_w	mass density of water [ML^{-3}]
θ	volumetric water content [$-$]
θ_r	residual volumetric water content [$-$]
θ_s	saturated volumetric water content [$-$]

Discretization of space and time

Ω	model domain
Γ	boundary of the model domain
d	spatial dimensionality
$\mathbf{x}$	spatial coordinates in a global, physical coordinate system
$\mathbf{X}$	spatial coordinates in a local or mixed coordinate system
t	time within the time interval $[0, T]$ with $T > 0$
$\mathcal{T}$	topological triangulation or subdivision in triangular prisms of Ω
E	an element within $\mathcal{T}$
F	a face within $\mathcal{T}$ being part of the boundary of an element
$\mathcal{E}$	set of elements
$\mathcal{F}$	set of faces
$\mathcal{N}$	set of nodes
$(\cdot)_I$	index of the inner domain
$(\cdot)_D$	index of Dirichlet boundaries
$(\cdot)_N$	index of Neumann boundies
$(\cdot)_R$	index of Robin boundaries
$\mathbf{n}$	unit normal vector pointing outwards
ν_F	unit normal vector on face F following a sign convention

List of Acronyms

CC-FVM	Cell-Centered Finite Volume Method
CG	Continuous Galerkin
DG	Discontinuous Galerkin
FD	Finite Differences
FDM	Finite Difference Method
FE	Finite Elements
FEM	Finite Element Method
FV	Finite Volumes
FVM	Finite Volume Method
HGS	HydroGeoSphere
$\mathcal{L}^2$	Lebesgue-space of square-integrable functions
MFEM	Mixed Finite Element Method
VC-FVM	Vertex-Centered Finite Volume Method

1 Introduction

Variably saturated subsurface flow is commonly simulated by substituting the extended law of Darcy into the continuity equation for water and assuming that the soil gas flows freely in the porous medium, leading to the Richards equation (Richardson, 1922; Richards, 1931; Celia et al., 1990). Accounting for soil hydraulic functions, the full Richards equation is a transient, non-linear partial differential equation. Under saturated conditions the Richards equation can be reduced to the linear groundwater flow equation, which is an elliptic (steady-state problem) or parabolic (transient problem) second-order differential equation of the hydraulic head (Bear and Cheng, 2010; Ngo et al., 2015; Matringe et al., 2006). Conservative solute transport is traditionally described by the advection-dispersion-equation (ADE) employing the velocity-field originating from the solution of the flow problem (Bear and Cheng, 2010; Kinzelbach, 1992). The ADE may numerically be solved by Eulerian methods, such as finite volume methods (FVM), continuous Galerkin (CG) methods, such as the standard Galerkin finite element method ($\mathcal{P}_1$ Galerkin FEM) (Bear and Cheng, 2010; Kinzelbach, 1992), or discontinuous Galerkin (DG) methods (Odsæter et al., 2017; Ngo et al., 2015), which are all computationally expensive and prone to numerical diffusion (Cirpka et al., 1999; Ngo et al., 2015). In advection-dominated transport, Lagrangian methods based on particle-tracking random walk are an attractive alternative and have repeatedly been used to compute travel times in engineering practice, and to analyze the impact of the spatial hydraulic-conductivity distribution on solute spreading (Kinzelbach and Ackerer, 1986; Tompson and Gelhar, 1990; Bellin et al., 1992, among others). Particle tracking has also been used to construct streamlines, on which efficient one-dimensional Eulerian transport schemes using a travel-time discretization can be applied (Crane and Blunt, 1999; Ginn, 2001; Atchley et al., 2013; Loschko et al., 2016, 2018).

The subsurface flow equation is commonly solved numerically for realistic modeling scenarios including heterogeneous boundary conditions and material properties. Most common numerical techniques include finite differences (FD), cell-centered finite volume methods (CC-FVM), vertex-centered finite volume methods (VC-FVM), continuous Galerkin methods (CG), like the standard $\mathcal{P}_1$ Galerkin finite element method (FEM), discontinuous Galerkin methods (DG), and mixed finite element methods (MFEM). Numerous adaptions, hybrids, combinations, and reformulations, like finite difference and finite volume methods formulated in finite-element terms, exist.

Standard continuous finite element and related methods are popular for subsurface flow simulations because they easily allow for unstructured grids and yield physically realistic, continuous hydraulic head fields. While cell-centered finite-volume formulations on an unstructured primal grid are possible (see e.g., Friis et al., 2009; Edwards and Zheng, 2010; Selzer and Cirpka, 2020), vertex-centered finite volumes (e.g., Huber and Helmig, 2000), or other finite-volume formulations on a dual grid (Edwards, 2002) are more common. Also, full material tensors can be incorporated in finite volumes, as an example, Edwards (2002) derived such techniques for a dual grid to unstructured triangles. Later Friis et al. (2009) and Edwards and Zheng (2010) extended this approach to the primal grid using a dual grid and transformations of coordinates

similar to those known from finite elements on subspaces, leading to more flexible schemes but also to a considerable computational overhead.

While continuous Galerkin methods, like the standard FEM and related techniques yield a continuous primary unknown, i.e., a physically realistic hydraulic-head field, this is not the case for finite-volume-type techniques, mixed finite element methods, and discontinuous Galerkin methods, which, in general, lead to a jump of the primary unknown on element boundaries (Odsæter et al., 2017; Selzer and Cirpka, 2020; Bastian and Rivière, 2003; Larson and Niklasson, 2004; Sun and Wheeler, 2006; Hoteit et al., 2002). However, in contrast to finite-volume-type techniques, standard FEM and related methods yield non-conforming velocity fields with the normal flux component being discontinuous across element boundaries yielding a physically unrealistic jump of the velocity on element boundaries (Odsæter et al., 2017; Selzer and Cirpka, 2020; Bastian and Rivière, 2003; Larson and Niklasson, 2004; Sun and Wheeler, 2006; Hoteit et al., 2002; Putti and Sartoretto, 2009), and mass is not locally conserved within the elements. While continuous Galerkin finite element methods approximate the unknown hydraulic-head field as a continuous function, they do not yield a conforming velocity field (Putti and Sartoretto, 2009; Ngo et al., 2015). If a particle is tracked on the basis of an element-wise approximated velocity field originating from $\mathcal{P}_1$ Galerkin FEM, the non-conforming property of the velocity field leads to severe numerical artifacts, possibly including particle stagnation because the approximated normal velocity component points into opposite directions at the two sides of an element interface (Putti and Sartoretto, 2009).

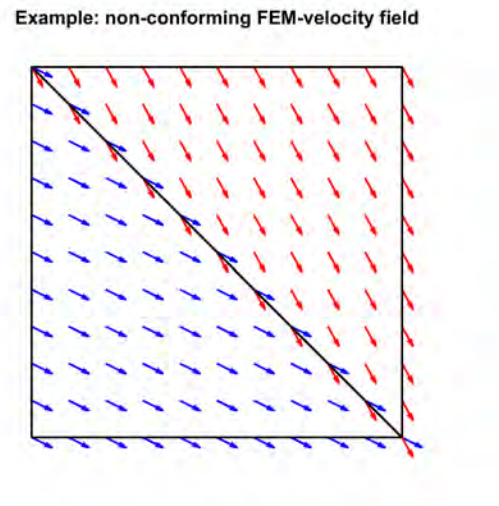

Figure 1: Exemplification of a non-conforming velocity field. A hypothetical particle exiting the left triangle (blue velocity vectors) will not be able to enter the right triangle (red velocity vectors) because both normal component of the depicted velocity field point in opposite direction. A solution for continuing a trajectory does not exist in this case, and the particle will stagnate on the interface between the triangles.

Figure 1 depicts an example of a non-conforming velocity field prohibiting to track a particle further than to the interface between the two triangles. A solution for continuing a trajectory does not exist, because the normal components of the two element-wise velocity fields point

outwards of each element, and therefore in opposite directions. Hence, the consistent application of particle-tracking methods to non-conforming velocity fields that may originate from standard FEM flow fields remains a problem. Several authors have emphasized that the $\mathcal{P}_1$ Galerkin FEM is locally mass-conservative, if appropriate grids and control volumes, which differ from the elements, or nodal fluxes are considered (Durlofsky, 1994; Hughes et al., 2000; Putti and Sartoretto, 2009). Especially, locally mass-conservative sub-control-volumes can be assigned to the element-wise approximation of the Darcy velocity field obtained by $\mathcal{P}_1$ Galerkin FEM for some sub-control-volumes on triangular elements carrying heterogeneous material coefficients (Putti and Sartoretto, 2009). According to the latter authors, such regions can be obtained by Voronoi tessellation or by patches around the nodes bounded by direct connections between the midpoints of edges of triangular elements sharing the respective node. They call these control volumes "internal A-cells". A delineation of control-volumes in analogy to vertex-centered finite volumes for triangular grids was proposed by Durlofsky (1994). In three dimensions, however, comparable control-volumes are not definable with similar ease, if hydraulic conductivity varies in space (Putti and Cordes, 1998; Putti and Sartoretto, 2009). Also, a conforming velocity interpolation within polygonal control volumes with a large number of faces, that may result from the reconstruction of Durlofsky (1994) or from vertex-centered finite volumes, can be non-trivial. That is, while local mass conservation in the elements is a necessary condition for reliable particle tracking, it may not be sufficient. In any case, considering the elements of a $\mathcal{P}_1$ Galerkin FEM grid itself, fluid mass is not conserved, and the normal component of the velocity is discontinuous, experiencing a jump on element-interfaces, while the primary unknown is continuous (Putti and Sartoretto, 2009; Scudeler et al., 2016).

Overall, one may conclude that standard FEM is the method of choice, if the primary interest lies in the hydraulic heads, while cell-centered finite-volume-type techniques are favorable, if the primary interest lies in the velocities.

1.1 Postprocessing of Non-Conforming Velocity Fields

Particle tracking represents an accurate and consistent method for solving solute transport only if it is based on a conforming velocity field, which implies that the normal component of the flux is continuous on element boundaries and mass is conserved in the elements. Such velocity fields may originate from a cell-centered finite volume method (Loschko et al., 2016; Pollock, 1988), a mixed finite element method (Juanes and Matringe, 2009; Matringe et al., 2006; Putti and Sartoretto, 2009), or from an $\mathcal{H}(div)$-projection of a non-conforming velocity field (Bastian and Rivière, 2003; Cordes and Kinzelbach, 1992; Kees et al., 2008; Larson and Niklasson, 2004; Odsæter et al., 2017; Selzer and Cirpka, 2020; Sun and Wheeler, 2006; Vidotto et al., 2018). Note that for standard discontinuous Galerkin methods, an $\mathcal{H}(div)$-projection can easily be defined locally avoiding the necessity of solving another global system of equations (Bastian and Rivière, 2003; Vidotto et al., 2018) although the normal component of the flux is not continuous on element boundaries. This is so because in the standard discontinuous Galerkin method the average normal fluxes on the element faces are locally conservative facilitating an element-wise

algorithm (Bastian and Rivière, 2003). However, the discontinuous Galerkin method can be considered a special case in this regard.

Several postprocessing techniques have been developed to obtain conforming velocity fields from standard FEM solutions. These techniques have allowed to simulate solute transport by Godunov type finite volume methods (Scudeler et al., 2016), discontinuous Galerkin methods (Odsæter et al., 2017), or particle tracking (Cordes and Kinzelbach, 1992; Selzer and Cirpka, 2020; Selzer et al., 2021). Simulations were either conducted on the primal grid (Kees et al., 2008; Larson and Niklasson, 2004; Odsæter et al., 2017; Selzer et al., 2021; Selzer and Cirpka, 2020; Sun and Wheeler, 2006) or on a dual grid (Cordes and Kinzelbach, 1992). While yielding conforming velocity fields, many existing postprocessing methods either face particular restrictions or induce numerical artifacts when applied to velocity fields that exhibit strong spatial variations, which is the case for highly heterogeneous porous media among others (Selzer and Cirpka, 2020; Schiavazzi, 2013; Odsæter et al., 2017).

Cordes and Kinzelbach (1992) introduced a scheme to reconstruct a conforming velocity field on linear triangular and bilinear quadrilateral Galerkin finite elements. It is based on the mass-conservation property of the internal A-cells (Cordes and Kinzelbach, 1992; Putti and Sartoretto, 2009). Considering the total fluxes on the boundaries of the internal A-cell, a system of equations can be set up for the total fluxes on all its inner boundaries (Cordes and Kinzelbach, 1992). This system has N equations with N variables, in which N is the number of inner edges of the internal A-cell. However, the circular structure causes the system of equations to be underdetermined. For regularization, Cordes and Kinzelbach (1992) set the constraint that the rotation of the hydraulic gradient within the elements must be zero. A direct extension of the method of Cordes and Kinzelbach (Cordes and Kinzelbach, 1992) to three dimensions is impossible, as there are 1.5 times more unknowns than equations related to internal nodes for tetrahedra, resulting in a highly underdetermined system of equations (Cordes and Kinzelbach, 1992; Putti and Sartoretto, 2009; Selzer and Cirpka, 2020).

Larson and Niklasson (2004) presented an approach to construct conforming velocity fields by postprocessing of $\mathcal{P}_1$ Galerkin FEM solutions that was originally defined for the case of a constant isotropic hydraulic conductivity. Sun and Wheeler (2006) introduced a similar, element-wise approach of flux correction to obtain local mass conservation and continuous normal components of fluxes on element boundaries for non-conforming velocity approximations, originating from continuous Galerkin finite elements (Povich et al., 2013; Scudeler et al., 2016). The approach of Larson and Niklasson (2004) and Sun and Wheeler (2006) is based on a discontinuous enrichment of the velocity field by applying a piecewise constant correction term which is added to the non-conservative, element-wise velocity approximation of the FEM on the element interfaces (Odsæter et al., 2017). The postprocessing techniques of Larson and Niklasson (2004) and Sun and Wheeler (2006) are defined as a local and a global method. The local method is particularly appealing because no global system of equations has to be solved (Larson and Niklasson, 2004; Scudeler et al., 2016). An adapted formulation applied to the Richards equation for variably saturated flow is given by Kees et al. (2008) and further employed and investigated

by Scudeler et al. (2016). However, the method of Larson and Niklasson (2004) and Sun and Wheeler (2006) was only derived and numerically tested for the Poisson equation (Larson and Niklasson, 2004) implying a steady-state, saturated flow and homogeneous material coefficients throughout the domain. An application to heterogeneous porous media and non-linear flow laws like the Richards equations may lead to pronounced numerical artifacts (Schiavazzi, 2013; Selzer and Cirpka, 2020) like spurious rotation, or regions of anomalously high velocities in low-flow regions. Artificial rotation has also occasionally been observed for some formulations of mixed finite element methods (Hoteit et al., 2002).

An advantage of the local variant of the flux correction proposed by Larson and Niklasson (2004) and Sun and Wheeler (2006) is, that it is parallelizable (Larson and Niklasson, 2004; Scudeler et al., 2016). However, numerical experiments conducted by Schiavazzi (2013) indicate that numerical errors are non-negligible for this scheme, if hydraulic conductivity strongly varies in space even in the isotropic case. Schiavazzi (2013) shows that the absolute values of element-wise constant Darcy-velocities can be orders of magnitude different in the postprocessed solution compared to the initial estimate of a velocity-space based on a $\mathcal{P}_1$ Galerkin FEM. Considering the particle tracks presented by Schiavazzi (2013) it appears, that also some artificial rotation is numerically introduced in the postprocessed solution. This might also be the reason why neither most of the particle tracks evaluated by Schiavazzi (2013) cross significantly zones of low permeability nor concentrations evaluated by Scudeler et al. (2016) enter zones of lower permeability to a significant extent, although Schiavazzi (2013) reports that the absolute magnitude of the element-wise postprocessed velocities can be orders of magnitude higher in low conductivity zones than those estimated on the basis of a $\mathcal{P}_1$ Galerkin FEM.

Recently Odsæter et al. (2017) presented an adapted scheme to those of Larson and Niklasson (2004) and Sun and Wheeler (2006), in which the residual between an element-wise given non-conforming velocity approximation and a conforming velocity solution is minimized with respect to a weighted $\mathcal{L}^2$-norm imposing local mass-conservation. Odsæter et al. (2017) introduce weighting factors equal to the inverse harmonic mean hydraulic conductivity on the element faces such that the normal component of the gradient rather than the Darcy-velocity is considered on element-boundaries. This leads to very good results also in heterogeneous domains, this choice of weighting factors apparently avoids the problems of the flux corrections applied and evaluated by Schiavazzi (2013) and Scudeler et al. (2016). However, such a scheme is not parallelizable in an easy manner anymore leading to a global system of equations.

1.2 Particle Tracking in Numerical Models

Particle-tracking methods require velocity fields generated by a preceding solution of the flow equation. Especially for advection-dominated transport, these methods represent an attractive alternative to solving the advection-dispersion equation by Eulerian methods because they do not suffer from the drawbacks of numerical dispersion or spurious negative concentrations (Cirpka et al., 1999, 2015; Crane and Blunt, 1999). Neglecting transverse dispersion, particle tracking replaces an explicit three-dimensional description of solute transport by a one-dimensional rep-

resentation employing travel time (Atchley et al., 2013; Crane and Blunt, 1999; Ginn, 2001; Loschko et al., 2016; Shapiro and Cvetkovic, 1988). It yields an easy to compute approximation of flow paths and travel times.

Pollock (1988) presented a particle-tracking method for cell-centered finite volumes on quadrilaterals and cuboids, which has been extended to deformed quadrilaterals and deformed hexahedra, eventually being split into tetrahedra, for non-divergent flow by Cordes and Kinzelbach (1992). Pollock's method (Pollock, 1988) is well established in groundwater hydrology, it has been classified as semi-analytical because it uses an analytical solution for particle trajectories based on a numerical approximation of the velocity field. It computes the exit-point of a particle on an element boundary or within the element from its entry-point by evaluating the fluxes over the volume boundaries, assuming that the normal component of the velocity vector on each volume-face is constant and each velocity component varies linearly in its own direction within the cell (Matringe et al., 2006; Pollock, 1988). This assumption about the velocity field corresponds to the lowest-order admissible velocity approximation (Matringe et al., 2006; Juanes and Matringe, 2009). Such a velocity field is in the lowest-order Raviart-Thomas-Nédélec-space ($\mathcal{RTN}_0$), which is also used for the velocity approximation of lowest-order mixed (hybrid) finite element methods (Raviart and Thomas, 1977; Nédélec, 1980, 1986; Matringe et al., 2006; Putti and Sartoretto, 2009; Bahriawati and Carstensen, 2005). Higher-order schemes are possible, but only seldom used in the simulation of groundwater flow and solute transport (Brezzi et al., 1985, 1987; Matringe et al., 2006; Juanes and Matringe, 2009; Vidotto et al., 2018). It is important to note that semi-analytical particle tracking is seldom applied outside of the field of groundwater hydrology. Numerical particle-tracking schemes using an integration procedure like the well established fourth-order Runge-Kutta method for solving initial value problems are more common (see e.g., Matringe et al., 2006; Juanes and Matringe, 2009). This has several reasons. First, in higher order velocity spaces, which are essentially all admissible velocity spaces other than the lowest-order Raviart-Thomas-Nédélec-space, the spatial dimensions cannot be decoupled, therefore, a simple closed-form solution for the trajectory within an element is not derivable. Furthermore, even in the lowest-order Raviart-Thomas-Nédélec-space, the spatial dimensions cannot be decoupled for deformed non-simplical elements, like deformed triangular prisms, prohibiting again a simple closed-form solution for the particle trajectory in physical, i.e. global, and local coordinates. However, approximations can be undertaken which at least give an easy measure of the average numerical error for an approximate analytical solution on a deformed non-simplicial element, if the solution for the non-deformed non-simplicial element is applied in an appropriate coordinate system to the deformed element. The alternative would be to employ a numerical integration procedure for approximating the trajectory within the deformed non-simplicial element, which also comes with an approximation error. However, semi-analytical particle tracking is very quick and stable, as the exit location of a particle can be computed in one step from the entry point and the element-wise velocity field, avoiding multiple steps required by a numerical approximation procedure, and avoiding possible overshooting of a particle over an element boundary, which is a common artifact in numerical integration. Especially for catchment-scale simulations and for embedding particle tracking in a Monte-Carlo type of

approach, semi-analytical particle tracking is desirable, as it drastically reduces the computation time, and avoids spurious numerical artifacts. Moreover, at least for a velocity field in lowest-order Raviart-Thomas-Nédélec-space on simplices and non-simplicial elements a closed analytical solution for the particle trajectories is possible for the non-deformed element. In this case, the trajectories are exact, with no numerical approximation error other than the one inherent in the velocity field, which is superior to any numerical integration procedure for computing trajectories.

1.3 Contribution of this Thesis

This work describes numerical methods and algorithms which are either already well known and established, or, to the best of my knowledge, new developments or extensions of existing methods. For clarification, the three major novel contributions of this thesis are highlighted in the following. Smaller novel contributions include a new analytical solution for groundwater flow on a rectangle with inflow and outflow windows of varying size including anisotropy as a ratio derived by Olaf A. Cirpka (see section 3.1.3) and an optimal strategy for splitting grids of structured cubes into conforming grids of tetrahedra such that the effects of grid orientation are minimized in lowest-order velocity approximations, like lowest-order FVM (or MFEM) discrectizations, developed by the author of this thesis (see section 3.1.4). The author of this thesis has also developed different approaches to find particles in grids of simplices, and triangular prisms based on the concept of barycentric coordinates (see section 2.6.5). However, barycentric coordinates are a well-known canonical approach for expressing coordinates in a simplex, among others often employed in computer graphics applications.

All other theory, numerical methods and algorithms described in this thesis are already well established, or sometimes just put differently. Formally, this thesis is a monography. However, the largest part of its content has already been published (Selzer and Cirpka, 2020; Selzer et al., 2021).[1] The main novel contributions are wrapped up in the following.

The $\mathcal{RTN}_0$-projection

The first major novel contribution is a postprocessing scheme, which is a novel formulation of an $\mathcal{RTN}_0$-projection employing concepts similar to those described by Odsæter et al. (2017). Our $\mathcal{RTN}_0$-projection maps a non-conforming, element-wise approximated Darcy velocity field of a standard Galerkin finite-element solution of the groundwater flow equation, or, in an extension, a finite-element-type primal solution of Richards' equation, representing both non-mass-conservative flow in the elements, onto a conforming velocity field in lowest-order Raviart-Thomas-Nédélec space. The target velocity space ensures continuity of the normal velocity component on element-boundaries, zero-divergence of the element-wise velocity field for non-divergent flow, and element-wise mass conservation. Implemented as a postprocessing code, the

[1] Because the author of this thesis is not the sole author of the publications: Selzer, P. and Cirpka, O. A. (2020): Postprocessing of standard finite element velocity fields for accurate particle tracking applied to groundwater flow. *Computational Geosciences* 24(4):1605–1624 & Selzer et al. (2021): Finite-volume flux reconstruction and semi-analytical particle tracking on triangular prisms for finite-element-type models of variably-saturated flow. *Advances in Water Resources* 154:103944 the generic "we" is used instead of "I" in the following.

$\mathcal{RTN}_0$-projector is formulated such that it can be coupled to any FEM code, but it comes at the costs that a new global system of equations has to be solved, which is a saddle-point problem of a size comparable to that of mixed finite elements. The key of the approach is that we minimize differences in the hydraulic gradient at the element centroids between the finite-element-type primal solution of hydraulic heads and the target $\mathcal{RTN}_0$ velocity field, subject to the constraint that the $\mathcal{RTN}_0$ velocity field is element-wise mass-conservative. We consider the residuals of the hydraulic gradient in the cell-center rather than that of the Darcy velocity because the hydraulic gradient varies much less than the Darcy velocity if the hydraulic conductivity field is heterogeneous, or flow is unsaturated implying heterogeneous relative permeabilities in space and time.

Moreover, such a minimization procedure should guarantee that the postprocessed velocity field has the same order of convergence as the original velocity approximation. In this regard, our postprocessing scheme is equivalent to the projection presented by Odsæter et al. (2017), who give a thorough analysis of the convergence behavior applicable also to our scheme. We found that a projection for the groundwater flow equation, in which residuals of the Darcy velocity were minimized, comparable to flux corrections formerly applied by Schiavazzi (2013) and Scudeler et al. (2016), introduced severe numerical artifacts because meeting the Darcy velocities in low-conductivity regions was less important in the optimization procedure than meeting those in high-conductivity regions, resulting in velocities of too high magnitude, and erroneous rotation and even reversal of the flow direction in zones of low hydraulic conductivity. Our $\mathcal{RTN}_0$-projection is reasonably applicable to any finite element method yielding non-conforming element-wise velocity fields. It can, in principle, be applied to various element types also in higher dimensions. The basic theory is given for simplices in two and three spatial dimensions, i.e., triangles and tetrahedra, for the groundwater flow equation, which facilitates a fully consistent formulation. However, the extension to finite-element-type primal solution of Richards' equation is outlined, too.

The drawback of these approaches is that an additional global system of equations has to be solved, which is a saddle-point problem of larger order than the initial FEM discretization, implying an increased computational effort and restrictions in the choice of the solver for the solution of the system of linear equations (Selzer and Cirpka, 2020). While the numerical results of Odsæter et al. (2017) and Selzer and Cirpka (2020) are very accurate, Selzer and Cirpka (2020) showed that such a postprocessing scheme may still induce numerical artifacts like artificial rotation, if velocities vary by several orders of magnitude in regions of similar hydraulic conductivity. Also, best results of the $\mathcal{RTN}_0$-projection will be achieved for saturated single-phase-flow in porous media on simplices. The $\mathcal{RTN}_0$-projection is described by Selzer and Cirpka (2020).

The FVM flux reconstruction

The second major novel contribution of this work is an alternative finite-volume based reconstruction of fluxes on the primal grid that does not directly project a finite-element-type velocity field onto a conforming velocity field. Our approach is based on a lowest-order cell-centered finite

volume method, requiring to solve a global system of linear equations that is strictly positive definite and whose size is on the order of the number of elements. Because lowest-order finite volumes used here are numerically very stable, the flux reconstruction does not introduce severe numerical artifacts, such as artificial rotation or anomalously high velocities in regions of low hydraulic conductivity.

Solving a global system of equations within a velocity-postprocessing scheme is computationally more beneficial when the flow equation is either transient or non-linear, such as Richards' equation. In that case, postprocessing requires solving only a single system of linear equations for each time at which the velocity is to be evaluated. This system of equations resembles that of a discretized linear steady-state flow equation, in which changes in storage may appear as a source/sink term, and the non-linear dependence of relative permeability on pressure head has already been accounted for in the preceding flow simulation. Postprocessing with a global system of equations remains computationally much faster than rerunning a full, transient model solving the non-linear Richards equation by a method that yields a conforming velocity field to begin with. The FVM flux reconstruction is described by Selzer et al. (2021).

Analytical solutions for divergent particle trajectories in $\mathcal{RTN}_0$-space on simplices

The third major contribution is a set of analytical solutions for semi-analytical particle tracking for simplices, and superposition of simplices, exemplified for triangular prisms, which make use of a conforming and element-wise mass conservative velocity field in $\mathcal{RTN}_0$-space. The framework of our particle-tracking scheme is analogous to Pollock's method (Pollock, 1988) and described by Selzer and Cirpka (2020) for simplices. It is defined in global, i.e., physical coordinates for triangles and tetrahedra, and in a mixed coordinate system for triangular prisms (Selzer et al., 2021). Also, the definition in local coordinates is sketched. Note that only the analytical solutions for divergent flow on simplices are genuinely novel contributions. The solutions for non-divergent flow are based on established techniques computing vectorized line-line and line-plane intersection. These solutions are well established in geometry and related fields like computer graphics, and only put here in the new context of particle tracking in subsurface hydrology.

1.4 Outline of this Thesis

The remaining chapters of this thesis are structured as follows:

Chapter 2 (Theory and Methodology) describes the governing equations, introduces the relevant set notation, gives a brief overview over the relevant function spaces, and provides a detailed description of the numerical methods and analytical solutions used and developed. This includes numerical methods for obtaining primary solutions, the two flux postprocessing schemes, namely the $\mathcal{RTN}_0$-projection and the FVM flux reconstruction, as well as the element-wise analytical solutions for the particle trajectories in divergent and non-divergent flow on simplices and triangular prisms, as an example of a superposition of simplices, in global, mixed, and local coordinates. Finally, the overall particle-tracking algorithm, which is an adaptation of the one

by Pollock (1988) is described as well as some implementation details.

Chapter 3 (Numerical Results) depicts numerical results for the $\mathcal{RTN}_0$-projection using different bench-mark cases for steady-state saturated flow on simplices. Empirical consistency and convergence is shown for an example case including hydraulic anisotropy. Moreover, the capabilities and drawbacks in application of the $\mathcal{RTN}_0$-projection are shown and discussed based on the consistency results and an application to variably saturated flow on triangular prisms. Subsequently, numerical results of the FVM flux reconstruction are shown and discussed for a bench-mark case as well as for a model of a sub-catchment in south-western Germany. To demonstrate the potential of the FVM flux reconstruction for catchment-scale subsurface flow-and-transport modeling, we apply it to a finite-element-type model on an unstructured grid consisting of triangular prisms that discretize the sub-catchment.

Both flux postprocessing schemes are accompanied with an analysis of according particle trajectories based on the developed semi-analytical scheme. Throughout this thesis, HydroGeoSphere (Aquanty, Inc., 2015; Therrien and Sudicky, 1996) is used to solve the variably saturated (or saturated) flow equation and to generate the primal solutions of the hydraulic head and relative permeability fields required for the $\mathcal{RTN}_0$-projection and for the FVM flux reconstruction. For groundwater flow we employ a standard Galerkin FEM, whereas we use the finite difference method in finite-element terms of HydroGeoSphere for computing variably saturated flow.

Chapter 4 (Conclusions and Outlook) discusses the major findings of this thesis, and highlights strengths as well as weaknesses of the different approaches. Here also some general findings are discussed having consequences not only for the methods described in this thesis but also for other numerical methods and postprocessing methods of non-conservative fluxes. An outlook is given including possible extensions and other approaches.

2 Theory and Methodology

2.1 Model Equations

Subsurface flow is governed by the interaction of pressure gradients, gravitational and capillary forces, and friction at pore walls. Porous media and natural porous formations are heterogeneous regarding the topological arrangements of potentially fluid-bearing pores, such that the real structure of porous-media flow cannot be resolved. Because of this, balance equations for a control volume are commonly formulated employing spatially averaged quantities and effective parameters. For sufficiently low Reynolds numbers the Darcy equation, which postulates a linear dependence of the flow velocity on the total-potential gradient, is valid. Combined with the continuity equation, it yields the common groundwater flow equation, the Richards equation, the multiphase-flow equations, and related transport equations in porous media. Throughout this thesis, we focus on the groundwater flow equation, the Richards equation, and the transport equation expressed in a Lagrangian formulation via particle tracking being all canonical model equations (Bear and Cheng, 2010; Kolditz, 2002).

2.1.1 Groundwater Flow Equation

We consider a bounded model domain $\Omega \subset \mathbb{R}^d$ with dimensionality d and the spatial coordinates $\mathbf{x} \in \Omega$, as well as the time interval $[0, T]$ with $T > 0$. Then, without further specifications of boundary and initial conditions, the groundwater flow equation reads as:

$$S_0 \frac{\partial h}{\partial t} - \nabla \cdot (\mathbf{K} \nabla h) = f, \quad (\mathbf{x}, t) \in \Omega \times [0, T], \tag{1}$$

where $\mathbf{K}$ [LT^{-1}] is the symmetric, positive-definite hydraulic-conductivity tensor, h [L] is the hydraulic head, f [$\mathrm{L^d L^{-d} T^{-1}}$] is a volumetric source/sink term per unit volume, t [T] is the temporal coordinate, and S_0 [L^{-1}] is the specific storage coefficient including the compressibility of the pore space and the water itself due to water pressure:

$$S_0 = \frac{\partial n_e}{\partial h} + \frac{n_e}{\rho_w} \frac{\partial \rho_w}{\partial h}, \tag{2}$$

where $n_e \in]0, 1[$ [$-$] is the effective porosity, and ρ_w [ML^{-3}] is the mass density of water.

For the $\mathcal{RTN}_0$-projection, we only consider subsurface flow with dimensionality $d \in \{2, 3\}$. This is because in one dimension the finite-element velocity field is already conforming, and if the grid is equally spaced, a finite-element formulation even reduces to a standard finite-difference approximation and to a finite-volume discretization employing an arithmetic mean of hydraulic conductivity for approximating the fluxes on the element boundaries (Celia et al., 1990). For an irregularly spaced grid equivalent finite-difference and finite-volume formulations can be found, too. The possibility of an equivalent finite-volume/finite-difference formulation also implies the possibility of dividing the one-dimensional finite-element formulation uniquely in mass-conservative sub-control volumes making an $\mathcal{RTN}_0$-projection unnecessary. Spatial

dimensions higher than $d = 3$ are also not considered because of the macro-physical spatial structure of our world and the assumptions of Newtonian mechanics.

For the $\mathcal{RTN}_0$-projection we only consider examples of variably saturated flow at steady state. However, temporal changes can be considered as an additional divergence term, and lumped into the source/sink term in a postprocessing scheme. If the groundwater flow equation is considered at steady state, the temporal derivative is set to zero, and the storage term drops out. Furthermore, we consider the domain boundary, $\partial\Omega = \Gamma$, which is subdivided into a Dirichlet boundary, Γ_D, and a Neumann boundary, Γ_N, such that $\partial\Omega = \Gamma_D \cup \Gamma_N$, and $\Gamma_D \cap \Gamma_N = \emptyset$.

Then, the elliptic steady-state groundwater flow equation is:

$$-\nabla \cdot (\mathbf{K}\nabla h) = f, \quad \text{in} \quad \Omega, \tag{3}$$

subject to the following boundary conditions throughout the following analysis:

$$\begin{aligned} h &= \hat{h}_D, && \text{on} \quad \Gamma_D, \\ \mathbf{n} \cdot (-\mathbf{K}\nabla h) &= \mathbf{n} \cdot \mathbf{q}_N, && \text{on} \quad \Gamma_N, \end{aligned} \tag{4}$$

where $\mathbf{q}_N$ is a specified volumetric flux density, and $\mathbf{n}$ is the unit normal vector pointing outwards of Ω. If the material properties are locally isotropic, the hydraulic conductivity tensor reduces to $\mathbf{K} = k\mathbf{I}$, where k [LT^{-1}] is the direction-independent scalar hydraulic conductivity, and $\mathbf{I}$ is the identity matrix of order d. Furthermore, the specific discharge, or Darcy velocity, $\mathbf{q}$ [LT^{-1}], is defined by Darcy's law:

$$\mathbf{q} = -\mathbf{K}\nabla h, \tag{5}$$

and related to the average linear velocity, $\mathbf{v}$ [LT^{-1}], via $\mathbf{q} = n_e\mathbf{v}$.

2.1.2 Richards' Equation

The Richards equation can be formulated in three variants, either using the pressure head, the water content, or both as primary unknowns (Celia et al., 1990). In the mixed version the transient term is the temporal change of the water content, while the divergence term is formulated employing the pressure head. Throughout this thesis a modified variant of the mixed version is used employing the hydraulic head and the volumetric water content as primary unknowns. This facilitates continuous saturated-unsaturated simulations and a mass-conservative first-order time discretization (Richardson, 1922; Richards, 1931; Celia et al., 1990; Forsyth, 1991; Therrien and Sudicky, 1996). Richards' equation is approximated in a spatial domain $\Omega \subset \mathbb{R}^3$ with $\mathbf{x} \in \Omega$ being the spatial coordinates and the domain boundary $\partial\Omega = \Gamma$, and within the time interval $[0, T]$ with $T > 0$, where t is again the temporal coordinate:

$$\frac{\partial\theta(\psi)}{\partial t} + \nabla \cdot \mathbf{q} = f, \quad (\mathbf{x}, t) \in \Omega \times [0, T], \tag{6}$$

where ψ is the pressure head [L], $\theta(\psi)$ is the volumetric water content [$-$], t is again time [T],

f represents again a volumetric source/sink term expressed as a volume of fluid exchanged per unit time and unit bulk volume [T^{-1}], and $\mathbf{q}$ is the specific discharge [LT^{-1}], or Darcy velocity, defined by the extended form of Darcy's law:

$$\mathbf{q} = -k_r(\psi)\mathbf{K}\nabla(\psi + z) \qquad \text{with} \qquad h = \psi + z, \tag{7}$$

where $k_r(\psi)$ is the relative permeability [$-$], $\mathbf{K}$ remains the hydraulic-conductivity tensor [LT^{-1}], z is the elevation head [L], and h is the hydraulic head [L]. We use the parametrizations of Mualem (1976) and van Genuchten (1980) for the soil-retention curve and the dependence of relative permeability on effective saturation, and pressure head, respectively:

$$S_e(\psi) = \begin{cases} \left(1 + (|\alpha\psi|)^N\right)^{\frac{1-N}{N}}, & \text{if } \psi < 0, \\ 1, & \text{if } \psi \geq 0, \end{cases} \tag{8}$$

$$\theta(\psi) = \theta_r + S_e(\psi)\left(\theta_s - \theta_r\right), \tag{9}$$

$$k_r(\psi) = \sqrt{S_e(\psi)}\left(1 - \left(1 - S_e(\psi)^{\frac{N}{N-1}}\right)^{\frac{N-1}{N}}\right)^2, \tag{10}$$

where α [L^{-1}] and N [$-$] are empirical coefficients, S_e denotes the effective saturation [$-$], whereas θ_r and θ_s are the volumetric residual and saturated water contents [$-$], respectively. For transient flow, the following approximation can be used for the term representing the change in storage in Equation 6:

$$\frac{\partial\theta(\psi)}{\partial t} \approx \frac{\theta(\psi)}{\theta_s} S_s \frac{\partial\psi}{\partial t} + \frac{\partial\theta(\psi)}{\partial t}, \tag{11}$$

where S_s is the specific storage coefficient under variably saturated conditions [L^{-1}].

The initial and boundary conditions are:

$$\begin{aligned} \psi(\mathbf{x}, t = 0) &= \psi_0, && \text{in } \Omega, \\ \psi(\mathbf{x}, t) &= \psi_D, && \text{on } \Gamma_D, \\ \mathbf{n} \cdot \mathbf{q}(\mathbf{x}, t) &= q_N, && \text{on } \Gamma_N, \\ c_0 \cdot \psi(\mathbf{x}, t) + c_1 \cdot \mathbf{n} \cdot \mathbf{q}(\mathbf{x}, t) &= c_2, && \text{on } \Gamma_R, \end{aligned} \tag{12}$$

where ψ_0 is the initial pressure head in the domain, ψ_D is a fixed pressure head, $\mathbf{n}$ is the outer unit normal vector, q_N is a specified normal flux, and $c_0, c_1, c_2 \in \mathbb{R}$ are coefficients. Moreover, Γ_D and Γ_N are defined as above as Dirichlet and Neumann boundary conditions, respectively, and Γ_R is the part of the boundary with a Robin boundary condition. A Robin boundary condition is often also referred to as a third-type boundary condition. It can be written as a linear combination of the primary unknown and its derivative. Note that $\partial\Omega = \Gamma = \Gamma_D \cup \Gamma_N \cup \Gamma_R$, and that Γ_D, Γ_N, and Γ_R are pairwise disjoint.

2.1.3 Transport Equation and Particle Tracking

We consider the bounded model domain $\Omega \subset \mathbb{R}^d$ with dimensionality d, and the time interval $[0, T]$ with $T > 0$. The traditional advection-dispersion equation in an Eulerian formulation for a conservative, non-reactive solute can be written as:

$$\frac{\partial c}{\partial t} + \mathbf{v}\nabla c - \nabla \cdot (\mathbf{D}\nabla c) = 0, \quad (\mathbf{x}, t) \in \Omega \times [0, T], \tag{13}$$

where c [ML^{-3}] is concentration, $\mathbf{v} = \mathbf{q}/\theta$ [LT^{-1}] is the linear average velocity, also called seepage velocity, $\theta = \theta(\psi)$ is the volumetric water content, and $\mathbf{D}$ [$\mathrm{L}^2\mathrm{T}^{-1}$] is the dispersion tensor. For purely advective flow, Equation 13 reduces to the following equation, which is equal to the total differential:

$$\frac{dc}{dt} = \frac{\partial c}{\partial t} + \mathbf{v}\nabla c. \tag{14}$$

Hence, if we exchange concentration c by a particle p having the characteristics $\mathbf{v}_p = d\mathbf{x}_p/dt$, where $\mathbf{x}_p$ is the trajectory of p being the characteristic line, Equation 14 becomes $dc/dt = 0$, which is equal to solving the advection equation (Equation 14) for this particle in a Lagrangian perspective. Given a large number of particles, concentration can be replaced by particle density facilitating to solve Equation 14 in a Lagrangian way, and therefore, replacing an explicit Eulerian discretization in up to three spatial dimensions by a one-dimensional discretization employing travel time. It the transported quantity is water or any other fluid, one may also associate a volume of fluid to every particle.

As a result, for a given velocity field $\mathbf{v}(\mathbf{x}, t)$, particle trajectories $\mathbf{x}_p(t)$ and associated travel times τ are computed by:

$$\mathbf{x_p}(t) = \mathbf{x}_{p,t_0} + \int_{t_0}^{t} \mathbf{v}_p(\tau) d\tau, \tag{15}$$

$$\tau = \int_S \frac{1}{\|\mathbf{v}(\mathbf{s})\|_2} d\mathbf{s}, \tag{16}$$

where $\mathbf{x}_{p,t_0}$ is the vector of starting coordinates, S is the (curvilinear) trajectory, while s denotes all coordinates defining S, and $\mathbf{v}_p$ denotes the linear average velocity in the coordinates s, which determines the particle trajectory S. The Lagrangian perspective of the trajectory, $\mathbf{x_p}(t)$, can be linked to the Eulerian perspective by: $\mathbf{v}(\tau) = \mathbf{v}(S(\tau))$.

2.2 Spatial Discretization of the Domain and Set Notation

Let $\mathcal{T}$ be either a topological triangulation of Ω into triangles ($d = 2$), or tetrahedra ($d = 3$), or a subdivision of Ω in triangular prisms. Furthermore, we restrict ourselves to conforming grids, which implies that all boundaries of neighboring elements perfectly fit onto each other. The elements are denoted $E_i \in \mathcal{T}$, with $i = 1, 2, ..., N$ where N is the total number of elements. The elements are equipped with the index i when necessary. For denoting two neighboring elements a

second index j may be introduced, where $i \neq j$ holds. Furthermore, we use the indices I, D, N, R to denote the inner domain, the Dirichlet boundary, the Neumann boundary, and the Robin boundary, respectively. Let $\mathcal{N}$ be the set of nodes, then $\mathcal{N}_{E_i}$ is the set of nodes of element E_i, and $\mathcal{N}_I$, $\mathcal{N}_D$, $\mathcal{N}_N$, and $\mathcal{N}_R$ are the sets of nodes in the inner domain, at the Dirichlet, Neumann, and Robin boundary, respectively. Moreover, $\mathcal{E}_I$ is the set of inner elements, $\mathcal{E}_D$ is the set of elements sharing a Dirichlet boundary, $\mathcal{E}_N$ is the set of elements sharing a Neumann boundary, and $\mathcal{E}_R$ is the set of elements sharing a Robin boundary. The set of elemental edges ($d = 2$), or elemental faces ($d = 3$) is denoted $\mathcal{F}$ with $F_m \in \mathcal{F}$, for $m = 1, 2, ..., M$, with M being the total number of edges or faces. For the sake of simplicity, we will only use the term "face" in the following, meaning either edges ($d = 2$), or faces ($d = 3$). The elements are equipped with the index m when necessary. Furthermore, let $\mathcal{F}_I$ be the set of all interior faces, $\mathcal{F}_D$ is the set of faces on the Dirichlet boundary, Γ_D, and $\mathcal{F}_N$ is the set of faces on the Neumann boundary, Γ_N, whereas $\mathcal{F}_R$ is the set of faces being part of a Robin boundary, Γ_R. These face sets are pairwise disjoint. $\mathcal{F}_B = \mathcal{F}_D \cup \mathcal{F}_N \cup \mathcal{F}_R$ is the set of all faces on the boundary $\partial\Omega$ of the domain. The Neumann boundary is split in a no-flow boundary, $\Gamma_{N,0}$, an inflow-boundary, $\Gamma_{N,+}$, and an outflow-boundary, $\Gamma_{N,-}$, comprising associated face sets such that all boundary segments are pairwise disjoint, and $\Gamma_N = \Gamma_{N,0} \cup \Gamma_{N,+} \cup \Gamma_{N,-}$ holds. Furthermore, let $\mathcal{F}_E$ be the set of faces of the element E, such that $F \in \mathcal{F}_{E_i}$ is an individual face of the element E_i. Furthermore, $\mathcal{N}_E$ denotes the set of nodes belonging to an element, whereas $\mathcal{N}_F$ denotes the set of nodes belonging to an individual face. We further define $\mathbf{n}_E$ as the unit vector normal to the boundary of element E, which is pointing outwards. Additionally, we specify $\mathbf{n}_{E,F}$ as the unit normal vector pointing outwards of element E on a face F, whereas $\boldsymbol{\nu}_F$ is a unit normal vector on face F following a sign convention. We use $|\cdot|$ to denote the measure of a set, in particular $|E|$ denotes the area ($d = 2$) or the volume ($d = 3$) of an element, and $|F|$ denotes the length ($d = 2$), or the area ($d = 3$) of a face. The Euclidean norm of a vector is denoted by $\|\cdot\|_2$, while $\langle\cdot,\cdot\rangle$ is the standard inner product. Furthermore, $[\cdot]_F$ denotes the jump of a function on a face.

2.3 Piecewise Polynomial Spaces

For clarification purposes, we briefly introduce some relevant, fundamental function spaces skipping their subspaces refined by boundary conditions (Bahriawati and Carstensen, 2005; Ngo et al., 2015; Odsæter et al., 2017). In the following sections, we will only refer to the respective function spaces, if we believe this is absolutely necessary for proper understanding of computations. Let $\mathcal{P}_1$ be the space of continuous piecewise linear polynomials such that:

$$\mathcal{P}_1(\mathcal{T}) = \{u \in \mathcal{C}(\overline{\Omega}) : u|_E \in \mathcal{Q}_1^d, \forall E \in \mathcal{T}\}, \tag{17}$$

where $\mathcal{C}$ is the space of continuous functions, and $\mathcal{Q}_1^d$ denotes the tensor product of linear polynomial spaces in each spatial direction (Odsæter et al., 2017). The space $\mathcal{P}_1(\mathcal{T})$ is defined such that it contains the approximated solution of the hydraulic head h computed by the standard Galerkin FEM. Therefore, it follows that $[h]_F = (h|_{E_i})|_F - (h|_{E_j})|_F = 0$ for the hydraulic head on any face, $F = E_i \cap E_j$, shared by two neighboring elements E_i and E_j, while

$[\mathbf{q} \cdot \boldsymbol{\nu}_F]_F = (\mathbf{q}|_{E_i})|_F \cdot \boldsymbol{\nu}_F - (\mathbf{q}|_{E_j})|_F \cdot \boldsymbol{\nu}_F$ can be non-zero in a $\mathcal{P}_1$ Galerkin solution. That is, the hydraulic head is conforming and continuous on elemental boundaries, while the normal component of the element-wise evaluated specific discharge can be discontinuous, experiencing a jump on element boundaries, leading to a non-conforming specific discharge field.

In lowest-order Raviart-Thomas-Nédélec-space for simplices (Raviart and Thomas, 1977; Nédélec, 1980, 1986; Bahriawati and Carstensen, 2005; Rognes et al., 2009) a vector-valued function in $\mathbb{R}^d$, in our case the specific discharge, is considered element-wise such that:

$$\begin{aligned}\mathcal{RTN}_0 =&\{\mathbf{u} \in \mathcal{L}^2(\Omega, \mathbb{R}^d) : \forall E \in \mathcal{T}\, \exists \boldsymbol{a} \in \mathbb{R}^d\, \exists b \in \mathbb{R} \forall \boldsymbol{x} \in E,\, \mathbf{u}(\mathbf{x}) = \mathbf{a} + b\mathbf{x}, \\ &\text{and}\, \forall F \in \mathcal{F}_I,\, [\mathbf{u} \cdot \boldsymbol{\nu}_F]_F = 0\},\end{aligned} \tag{18}$$

where $\mathcal{L}^2(\Omega, \mathbb{R}^d)$ is the usual Lebesgue-space of square-integrable functions in $\mathbb{R}^d$ (e.g. Bahriawati and Carstensen, 2005; Rognes et al., 2009). Considering the definition of the $\mathcal{RTN}_0$-space in Equation 18, it is obvious, that not only the normal component of a vector-valued function is continuous on the faces, but also its divergence is element-wise constant and linear dependent on b, implying also that the divergence is element-wise zero for non-divergent flow. A triangular prism is a superposition of a triangle, which is the simplex in two dimensions ($d = 2$), and a line, which is the simplex in one dimension ($d = 1$). By this, the $\mathcal{RTN}_0$-space for a prism is simply given by superposition of the $\mathcal{RTN}_0$-space 18 with itself, given $d = 2$ in the horizontal, and $d = 1$ in the vertical. For simplicity, we refer in the following to the $\mathcal{RTN}_0$-space no matter, if a simplex, or a superposition of simplices, like a triangular prism, is meant.

In this thesis, an $\mathcal{RTN}_0$-projection is proposed, by which we intend to project an element-wise approximated specific-discharge field originating in our application from a $\mathcal{P}_1$ Galerkin FEM for saturated flow, and a finite difference method in finite-element terms for variably saturated flow, onto a conforming velocity field in $\mathcal{RTN}_0$-space. This can be expressed by the following map:

$$\pi^{RTN_0} : \mathcal{L}^2(\Omega, \mathbb{R}^d) \rightarrow \mathcal{RTN}_0. \tag{19}$$

2.4 Numerical Methods for Spatial Discretization

2.4.1 The Standard Galerkin Finite Element Method

The $\mathcal{P}_1$ Galerkin finite element method, also named standard Galerkin FEM, is a common numerical method to solve the groundwater flow equation (Equation 1) (Huyakorn and Pinder, 1983; Kinzelbach, 1992; Bear and Cheng, 2010). $\mathcal{P}_1$ Galerkin FEM yields a continuous approximation of the hydraulic-head field, and naturally handles unstructured grids and anisotropy. In the following, we repeat the standard finite element formulation for the groundwater flow equation (Equation 1) yielding a conforming hydraulic-head field, which is piecewise linear for simplices, and piecewise bi-linear for superpositions of two simplices, like quadrilaterals or triangular prisms. For superpositions of three simplices, like cubes, the solution is tri-linear and so on and so forth for arbitrary superpositions of simplices in multiple dimensions leading to multi-linearity. In the following, we give an element-wise description, an alternative node-wise

formulation is possible, too, and sketched in the following section 2.4.2 (see also Forsyth, 1991; Therrien and Sudicky, 1996) for a finite difference method in finite-element terms. In this section, we only describe a standard finite element formulation for the groundwater flow equation (Equation 1), and not for the Richards equation (Equation 6). The reason is that a stable numerical approximation of Richards' equation relies on full upstream weighting of relative permeability. In an element-wise formulation like the one below, a consistent formulation of upstream weighting is not possible in a straight-forward manner because we do not only consider the direct nodal connections, but also approximate the primary unknown continuously within the element to get a conforming solution of the hydraulic head. That is, within the element a fully consistent, full upstream weighting can not be guaranteed for discretizations in more than one spatial dimension, which can lead to instabilities, and non-convergence in practice. Moreover, the native averaging procedure in finite elements would be arithmetic averaging (Helmig and Huber, 1998), or centered differences, respectively, which, in principle, is second-order accurate, but it is also known to produce a non-physical oscillatory numerical behavior, if the solution is not smooth enough (Ippisch, 2001). This prohibits using arithmetic averaging, or a central difference scheme, respectively, in practice for solving Richards' equation.

Still, a finite-element discretization of Richards' equation in analogy to the one given below is possible. However, for the named reasons we opt to describe the discretization of Richards equation in finite-element terms in the following section 2.4.2 employing a nodal formulation, which allows an easy switch between a finite-element formulation, and a finite-difference formulation in finite-element terms. The latter again allows for a straight-forward, consistent upstream weighting because only the direct nodal connections are considered, and the hydraulic head is linearly interpolated only on these direct nodal connections, and not within the element itself (Therrien and Sudicky, 1996; Forsyth, 1991).

Considering the space $\mathcal{P}_1(\mathcal{T})$, the linear shape functions χ_i corresponding to the nodes i can be defined element-wise such that they sum up to unity at all points in all elements:

$$\sum_{i \in \mathcal{N}_E} \chi_i(\mathbf{x}) = 1, \quad \forall \mathbf{x} \in E \wedge \forall E \in \mathcal{T}, \tag{20}$$

where χ_i are the shape functions according to the space $\mathcal{P}_1(\mathcal{T})$ equaling 1 on node i and 0 on any other node $j \neq i$. We replace the hydraulic-head field $h(\mathbf{x})$, the source/sink term-field $f(\mathbf{x})$, and the specific storage coefficient, $S_0(\mathbf{x})$, by the approximate, piecewise linear functions $\hat{h}$, $\hat{f}$, and $\hat{S}_0$, respectively, such that:

$$h \approx \hat{h} = \sum_{i \in \mathcal{N}} h_i \chi_i, \quad f \approx \hat{f} = \sum_{i \in \mathcal{N}} f_i \chi_i, \quad S_0 \approx \hat{S}_0 = \sum_{i \in \mathcal{N}} S_{0,i} \chi_i, \tag{21}$$

where h_i is the hydraulic-head value, f_i is the source/sink term, and $S_{0,i}$ is the specific storage coefficient all at node i, respectively. Note that also standard finite element formulations exist, where the source/sink term is not linearly distributed, but just a lumped quantity on the nodes subject to multiplication with a weighting function only. We apply the concept of weak derivatives, and choose the weighting functions to equal the shape functions. This yields the

well-known standard Galerkin FEM discretization, which reads element-wise as:

$$\int_E \boldsymbol{\chi}_E^T \hat{\mathbf{S}}_\mathbf{0} \boldsymbol{\chi}_E \, d\mathbf{x} \frac{\partial \hat{\mathbf{h}}}{\partial t} + \int_E \nabla \boldsymbol{\chi}_E^T \mathbf{K}_E \nabla \boldsymbol{\chi}_E \, d\mathbf{x} \, \hat{\mathbf{h}} = \int_E \boldsymbol{\chi}_E^T \boldsymbol{\chi}_E \, d\boldsymbol{x} \, \hat{\mathbf{f}}, \quad \forall E \in \mathcal{T}, \tag{22}$$

where $\boldsymbol{\chi}_E$ are the elemental shape functions, $\mathbf{K}_E$ is the element-wise given hydraulic-conductivity tensor, $\hat{\mathbf{h}}$ is the vector of nodal hydraulic-head values of element E, $\hat{\mathbf{S}}_\mathbf{0}$ is the vector of nodal specific-storage coefficients, and $\hat{\mathbf{f}}$ is the vector of nodal source/sink strengths.

The weak form (see Equation 22) implies that derivatives do not need to exist everywhere in the domain, but only in an integrated sense. The concept of weak derivatives can be applied here, because every term is an integral term. Based on $\hat{\mathbf{h}}$, a unique, element-wise constant gradient can be computed on simplices, leading to an element-wise constant, non-conforming specific-discharge vector. This unique gradient can be obtained by evaluating the gradient of the shape functions multiplied with the vector of nodal hydraulic-head values. On non-simplicial elements the specific discharge vector varies within the element. The element-wise specific-discharge vector is approximated by:

$$\mathbf{q}_E^{FE} = -\mathbf{K}_E \nabla \boldsymbol{\chi}_E \hat{\mathbf{h}}, \quad \forall E \in \mathcal{T}. \tag{23}$$

2.4.2 Finite Difference Method in Finite-Element Terms

For obtaining finite-element-type primary solutions, throughout this thesis HydroGeoSphere is used (Therrien and Sudicky, 1996; Aquanty, Inc., 2015). HydroGeoSphere either solves the groundwater flow equation (Equation 1) or a modified Richards equation in mixed form (Equation 6) either using a standard Galerkin FEM, or a finite difference method formulated in finite-element-terms. Subsurface porous media flow can also be simulated as a dual continuum model, or be coupled with flow in fracture systems, or overland flow. However, we do not consider these different flow regimes, and the associated coupling approaches throughout this thesis, and solely focus on subsurface variably saturated flow. In the following, we describe the finite-element-type formulation used in HydroGeoSphere leading to a finite-difference formulation in finite-element terms. The following derivation roughly follows Forsyth (1991) and Therrien and Sudicky (1996). Note that the following derivation as well as description of the mathematical attributes differs significantly from the ones given in the HydroGeoSphere manual (Aquanty, Inc., 2015) at the current date. This is because an increased accuracy is desired by the author of this thesis.

Recall the linear shape functions, χ_i, used in the $\mathcal{P}_1$ Galerkin FEM (Equation 20), corresponding to the nodes i and summing up to unity everywhere in the domain. In analogy to the discretized quantities in Equation 21, the discretized volumetric water content can be formulated:

$$\theta(\psi) \approx \hat{\theta}(\psi) = \sum_{i \in \mathcal{N}} \theta_i(\psi) \chi_i, \tag{24}$$

where θ_i is the volumetric water content at node i. Note, that the discretized volumetric water content is set here for simplicity instead of the enlarged storage on the right-hand side of

Equation 11, by which the volumetric water content can be replaced for saturated-unsaturated simulations.

Note that an element-wise representation of a standard Galerkin finite element formulation is tedious as it remains unclear how a fully consistent weighting procedure of relative permeability not only between the nodes directly but also interpolated within the element should be expressed best. This thought directly leads to a nodal finite-element-type formulation. To avoid the problem of an inconsistent weighting of relative permeability within the elements, one may furthermore reduce the scheme to the direct nodal connections avoiding the cross terms arising in a full finite element formulation. Such a scheme is then a finite difference method formulated in finite-element terms facilitating a stable, and consistent solution of Richards' equation. Such a solution is continuous on lines, which are the direct nodal connections, embedded in three-dimensional or other-dimensional space. The drawback is that the solution within the elements remains undefined, it can only be computed in a postprocessing step using the shape functions of a full finite element formulation. Additionally, hydraulic conductivity can only be expressed as a diagonal tensor, and not as a full tensor, because there are no cross terms anymore in the numerical discretization. However, the advantage is that a stable solution of Richards' equation can be computed including consistent upstream weighting. The unavoidable backside of upstream weighting as a first-order accurate scheme is numerical smoothing of sharp fronts, and an increased sensitivity to grid orientation compared to schemes which are higher-order accurate (compare e.g., Ippisch, 2001). Moreover, one may evaluate hydraulic heads and Darcy velocities also within the element using the full finite element shape functions. However, such an approach is inconsistent with how the hydraulic heads are computed in the primary solution.

For formulating such a finite difference method in finite-element terms, we go one step back, and start with the basic weighted residual approach, employing the discretized quantities (see Equations 21 and 24):

$$\int_\Omega \mathbf{W} \frac{\partial \hat{\theta}(\psi)}{\partial t}\, d\mathbf{x} - \int_\Omega \mathbf{W} \nabla \cdot (k_r \mathbf{K} \nabla \hat{h})\, d\mathbf{x} = \int_\Omega \mathbf{W} \hat{f}\, d\mathbf{x}, \tag{25}$$

where $k_r = k_r(\psi)$, $\theta = \theta(\psi)$, and $\mathbf{W}$ is the weighting function. Note that Equation 25 is formulated for the whole computational domain Ω. In the Galerkin technique the weighting function equals the shape function:

$$\mathbf{W} = \boldsymbol{\chi}^T, \tag{26}$$

where $\boldsymbol{\chi}$ contains the linear (or bilinear, trilinear, etc.) shape functions in the domain Ω. Using the product rule of differentiation and the divergence theorem, Equation 25 can be reformulated yielding the weak form of Equation 6 for the elements. However, we still demand for weak derivatives, but follow another route for the finite difference method in finite-element terms. We first focus on the divergence term, and the Galerkin-type shape and weighting functions. We

apply the product rule of differentiation:

$$\int_{\Omega} \boldsymbol{\chi}^T \nabla(k_r \mathbf{K} \nabla \hat{h})\, d\mathbf{x} = \int_{\Omega} \nabla \left(\boldsymbol{\chi}^T (k_r \mathbf{K} \nabla \hat{h}) \right)\, d\mathbf{x} - \int_{\Omega} (\nabla \boldsymbol{\chi})^T (k_r \mathbf{K} \nabla \hat{h})\, d\mathbf{x}, \tag{27}$$

applying the divergence theorem to the first term on the right hand-side yields:

$$\int_{\Omega} \boldsymbol{\chi}^T \nabla(k_r \mathbf{K} \nabla \hat{h})\, d\mathbf{x} = \int_{\partial\Omega} \mathbf{n} \cdot \left(\boldsymbol{\chi}^T (k_r \mathbf{K} \nabla \hat{h}) \right)\, d\mathbf{s} - \int_{\Omega} (\nabla \boldsymbol{\chi})^T (k_r \mathbf{K} \nabla \hat{h})\, d\mathbf{x}, \tag{28}$$

where $\mathbf{n}$ is the outer unit normal vector on the domain boundary $\Gamma = \partial\Omega$. We require global mass conservation, therefore, we can set the boundary integral over $\Gamma = \partial\Omega$ to zero. This yields:

$$-\int_{\Omega} \boldsymbol{\chi}^T \nabla(k_r \mathbf{K} \nabla \hat{h})\, d\mathbf{x} = \int_{\Omega} (\nabla \boldsymbol{\chi})^T (k_r \mathbf{K} \nabla \hat{h})\, d\mathbf{x}. \tag{29}$$

So far, we just followed a standard procedure for deriving the standard Galerkin technique. Combining the rewritten divergence term with the Equations 25 and 26, and splitting the shape function from the approximate functions $\hat{h}$, $\hat{\theta}$, and $\hat{f}$, one can formulate an element-wise expression employing the nodal quantities $\hat{\boldsymbol{\theta}}$, $\hat{\mathbf{h}}$, and $\hat{\mathbf{f}}$ like for the groundwater flow equation (Equation 22). This would be the standard Galerkin formulation of the Richards equation in direct analogy to the standard Galerkin discretization of the groundwater flow equation (Equation 22). However, we do not want to follow this route for the already named reason that consistent upstream weighting of relative permeability is not possible for this formulation within the elements.

Now, we neither refer to the shape functions and the weighting functions as tuples of functions over the domain, $\boldsymbol{\chi}$, $\mathbf{W}$, nor as element-wise functions or matrices, $\boldsymbol{\chi}_E$, $\boldsymbol{W}_E$, anymore. We focus again on the nodal function, which is also known as "hat-function" (compare Figure 2b). Let i and j be indices ranging over all the nodes, then χ_i is the linear shape functions for node i, if the element is a simplex, which would be the "hat"-function in one or two dimensions. For multi-linear elements, the shape function would be the corresponding multi-linear function for node i. I.e. the shape function belonging to node i reaches over all elements sharing i. Then:

$$\begin{aligned} \chi_i &= 1, \text{ at node i,} \\ \chi_i &= 0, \text{ at all other nodes,} \\ \sum_j \chi_j &= 1, \text{ everywhere in the domain.} \end{aligned} \tag{30}$$

Regarding the node i and its volume of influence, we may reformulate the divergence term (Equation 2.4.2) for the nodes. By this we replace a formulation over the computational domain Ω by an integral restricted to the zone of influence of one node, which is the element-star, $\mathcal{E}_i$, consisting of all elements sharing node i. If we sum over all the resulting integrals, the result is the same as a formulation over Ω. Moreover, we will restrict ourselves in the following to the direct nodal connections between the node i and nodes j omitting the cross terms, being the connections only between nodes j for a node i. So far, relative permeability and hydraulic conductivity were

just assumed to be some functions in space $\mathbf{K} = \mathbf{K}(\mathbf{x})$, or in space and time $k_r = k_r(t, \mathbf{x})$. Both will now be restricted to the nodal values used for the flux approximation between the nodes i and j, which we will denote $\mathbf{K}_{ij}$ and $k_{r,ij}$, and which is a reduction of the standard Galerkin discretization to the direct nodal connections implying an according reduction of the shape function χ (and by this also of the weighting function) to the direct nodal connections. By this, a solution is not approximated in the elements, but the solution reduces to continuous solution on the direct nodal connections, which can be thought of a network of lines embedded in higher-dimensional space:

$$\int_{\Omega} (\nabla \boldsymbol{\chi})^T (k_r \mathbf{K} \nabla \hat{h}) \, d\mathbf{x} = \sum_{i \in \mathcal{N}} \int_{\mathcal{E}_i} \nabla \chi_i k_{r,ij} \mathbf{K}_{ij} \nabla \left(\sum_j h_j \chi_j \right) \, d\mathbf{x}, \tag{31}$$

where j can either range over all nodes of the domain, or only over the nodes being part of $\mathcal{E}_i$, with identical result. Note also that χ_i is the shape function in dimension d such that the according matrix-vector operations are valid. From the attributes of the shape function, we know that:

$$\chi_i = 1 - \sum_{j \neq i} \chi_j, \tag{32}$$

and

$$\nabla \chi_i = -\nabla \sum_{j \neq i} \chi_j. \tag{33}$$

We may now reformulate the Galerkin discretization of the divergence term focusing on one node i 31:

$$\begin{aligned} \int_{\mathcal{E}_i} \nabla \chi_i k_{r,ij} \mathbf{K}_{ij} \nabla \left(\sum_j h_j \chi_j \right) \, d\mathbf{x} &= \int_{\mathcal{E}_i} \nabla \chi_i k_{r,ij} \mathbf{K}_{ij} \nabla \left(\sum_{j \neq i} h_j \chi_j \right) + \nabla \chi_i k_{r,ij} \mathbf{K}_{ij} \nabla \left(h_i \chi_i \right) \, d\mathbf{x} \\ &= \int_{\mathcal{E}_i} \nabla \chi_i k_{r,ij} \mathbf{K}_{ij} \nabla \left(\sum_{j \neq i} h_j \chi_j \right) - \nabla \chi_i k_{r,ij} \mathbf{K}_{ij} \nabla \left(\sum_{j \neq i} h_i \chi_j \right) \, d\mathbf{x} \\ &= \int_{\mathcal{E}_i} \nabla \chi_i k_{r,ij} \mathbf{K}_{ij} \nabla \left(\sum_{j \neq i} \chi_j (h_j - h_i) \right) \, d\mathbf{x}. \end{aligned} \tag{34}$$

We will denote by η_i the set of nodes being part of the element-star, $\mathcal{E}_i$, excluding node i. Then, we reformulate the divergence term:

$$\int_{\mathcal{E}_i} \nabla \chi_i k_{r,ij} \mathbf{K}_{ij} \nabla \left(\sum_{j \neq i} \chi_j (h_j - h_i) \right) \, d\mathbf{x} = \sum_{j \in \eta_i} \int_{\mathcal{E}_i} \nabla \chi_i k_{r,ij} \mathbf{K}_{ij} \nabla \chi_j (h_j - h_i) \, d\boldsymbol{x}. \tag{35}$$

One may now formulate the fully discretized Richards equation using the reduced shape function consistent with the finite difference method in finite-element terms for the element-star belonging

to an internal node i:

$$\sum_{j\in\eta_i}\int_{\mathcal{E}_i}\chi_i\chi_j\,d\mathbf{x}\frac{\partial\theta_i}{\partial t}+\sum_{j\in\eta_i}\int_{\mathcal{E}_i}\nabla\chi_i k_{r,ij}\mathbf{K}_{ij}\nabla\chi_j\,d\boldsymbol{x}(h_j-h_i)=\sum_{j\in\eta_i}\int_{\mathcal{E}_i}\chi_i\chi_j\;d\boldsymbol{x}\,f_i\quad\forall i\in\mathcal{N}_I. \tag{36}$$

where $\mathbf{K}_{ij}$ is in practice obtained by averaging the element-wise hydraulic conductivity values. Note also, that $\mathbf{K}_{ij}$ cannot be a full tensor in a finite difference method in finite-element terms. Moreover, the principal directions of $\mathbf{K}$ should coincide with the direction of the lines connecting the nodes in a finite-difference approximation for most accurate results. For obtaining the hydraulic conductivity typically one would take a weighted arithmetic mean of the hydraulic conductivities associated to the elements sharing a node i. For such averaging procedures, HydroGeoSphere typically takes a weighted arithmetic average of hydraulic conductivity weighted by the volumes of the elements sharing node i, or the parts of the elements associated to node i delineated by a control volume around node i by making use of the centroid (compare Figure 2).This would be associated to an additional weighting factor $1/|\mathcal{N}_E|$, which is 1/3 for triangles, and 1/6 for triangular prisms for the elemental volumes, where the denominater equals the number of nodes per element. However, as this scaling factor is the same for all neighboring elements in a discretization of the same element-type these factors cancel out in a weighted arithmetic mean anyways.

The main advantage of formulation 36 is that full upstream weighting of relative permeability $k_{r,ij}$ can be formulated in a consistent manner:

$$k_{r,ij}=\begin{cases}k_{r,i}, & \text{if}\quad h_i\geq h_j\\ k_{r,j}, & \text{if}\quad h_j>h_i,\end{cases} \tag{37}$$

where $k_{r,i}$ and $k_{r,j}$ are the relative permeabilities associated to the nodes i and j for a specific element, respectively. Note also that relative permeability may differ for the same node in different elements, if neighboring elements have different unsaturated material properties.

Note also that for nodes at the boundary of the domain, $\mathcal{N}_B=\mathcal{N}\setminus\mathcal{N}_I$, Equation 36 is either superimposed with the boundary condition or replaced by the boundary condition depending on the type and the implementation of the different boundary conditions, and if another hydrosystem is considered as boundary, respectively.

Expression 36 is consistent with the finite-difference discretization in finite-element terms on a topological line-network. However, also inconsistent formulations exist, some of them leading to an increased stability of the numerical solution procedure. One of these inconsistent formulations leading to increased stability is mass lumping, which can also be applied in HydroGeoSphere. In classical mass lumping the storage matrix discretizing the transient term is simplified such that all entries of each row are summed up, and the result is put on the diagonal. This not only leads to a diagonal matrix but also avoids the linear distribution of the change in storage between the nodes consistent with the standard Galerkin discretization on a line-network. By this, all the change in storage is attributed to the nodes, which results in a similar storage matrix than in

finite-volume techniques. In fact, for the change in storage using mass lumping control volumes similar to those of vertex-centered finite volumes can be attributed. These control volumes are delineated by the connections between the midpoints of the direct nodal connections and the centroids of the elements. Mass lumping can be employed by HydroGeoSphere no matter if the subsurface flow equation is solved by standard Galerkin finite elements or by the described finite difference method in finite-element terms. An other simplification HydroGeoSphere employs, is that nodal sources and sinks are not linearly distributed, too, but lumped to the nodes as an average quantity evaluated in the described variant of vertex-centered-finite-volume-type control volumes. This is done in analogy to mass lumping of the storage matrix, and again identical to a vertex-centered finite-volume formulation. This means neither the transient term nor the sources and sinks are consistent with Galerkin type finite-difference discretization of the divergence term, in the finite difference formulation in finite-element terms. In fact, the approach for the transient term and for the source/sink term of HydroGeoSphere is to use a weighting function which is 1 for the control volume associated to node i, and 0 elsewhere, while still keeping a standard Galerkin discretization for the divergence term, though only considering the direct nodal connections.

However, it should be noted that such an inconsistent scheme leads to increased stability of the solution procedure because only the divergence term is approximated via a Galerkin discretization on the direct nodal connections, and full upstream weighting can be achieved for relative permeability. The explanations above are backed by Therrien and Sudicky (1996) and Aquanty, Inc. (2015), as well as by personal communication with René Therrien and own findings of the author of this thesis. Note also, that for solving the temporal derivative HydroGeoSphere uses an implicite Euler scheme. However, in our current description we focus on the spatial discretization because of its relevance for developing postprocessing techniques. The modified discretization of Richards' equation for node i solved by HydroGeoSphere could then be formulated using a discontinuous weighting function:

$$W_i^{B_i} = \begin{cases} 1, & \text{if } \mathbf{x} \in B_i \\ 0, & \text{if } \mathbf{x} \notin B_i \end{cases}, \tag{38}$$

where B_i is the box-like control volume associated to node i, which can be defined using the centroids of the elements, and the centroids of the internal faces in $\mathcal{E}_i$ for delineation. Furthermore, for a simplex the special relationship holds that the area ($d = 2$) or the volume ($d = 3$) of B_i equals to:

$$|B_i| = \sum_{E_j \subset \mathcal{E}_i} \frac{|E_j|}{|\mathcal{N}_{E_j}|}, \tag{39}$$

where j is an index ranging over the elements E_j being part of the element-star $\mathcal{E}_i$, and $\mathcal{N}_{E_j}$ is the set of nodes being part of E_j.

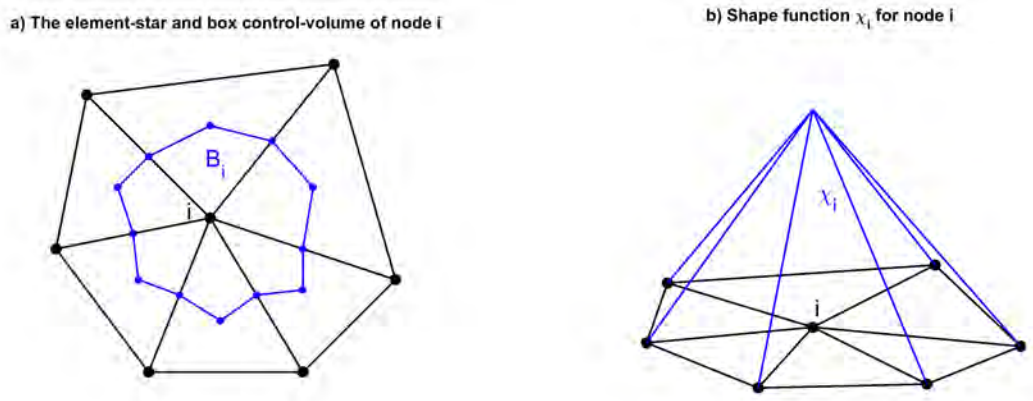

Figure 2: Depiction of the element-star $\mathcal{E}_i$, which is the union of all elements sharing node i, and imposed control-volumes and functions exemplified on triangles; a) $\mathcal{E}_i$ and its associated node-centered control volume B_i for the weighting function $W_i^{B_i}$ (Equation 38) applied to the transient term and the source/sink term in Equation 42 being one within B_i and zero elsewhere; b) Shape function χ_i for node i, note that the shape function equals the weighting function for the divergence term in Equation 42.

Then an inconsistent, but more stable discretization of Richards' equation including mass-lumping for the change in storage and a lumped source/sink term reads:

$$\sum_{j\in\eta_i}\int_{\mathcal{E}_i} W_i^{B_i}\, d\mathbf{x}\frac{\partial\theta_i}{\partial t} + \sum_{j\in\eta_i}\int_{\mathcal{E}_i} \nabla\chi_i k_{r,ij}\mathbf{K}_{ij}\nabla\chi_j\, d\boldsymbol{x}(h_j - h_i) = \sum_{j\in\eta_i}\int_{\mathcal{E}_i} W_i^{B_i}\, d\boldsymbol{x}\, f_i, \quad \forall i \in \mathcal{N}_I. \tag{40}$$

where $\mathcal{E}_i$ is still the union of the elements which share the node i. Recall also that:

$$\int_{\Omega} W_i^{B_i} d\mathbf{x} = |B_i|\,, \tag{41}$$

which reduces Equation 42 to:

$$\frac{\partial\theta_i}{\partial t}|B_i| + \sum_{j\in\eta_i}\int_{\mathcal{E}_i} \nabla\chi_i k_{r,ij}\mathbf{K}_{ij}\nabla\chi_j\, d\boldsymbol{x}(h_j - h_i) = |B_i| f_i, \quad \forall i \in \mathcal{N}_I. \tag{42}$$

2.4.3 A Cell-Centered Finite Volume Method for Unstructured Grids

In this section, we present a cell-centered finite volume method for unstructured grids, either being a topological triangulation (i.e., composed out of simplices like triangles or tetrahedra), or an other subdivision of the model domain into polyhedra like triangular prisms. Note that, throughout this thesis, we assume conforming grids, where all boundary faces fit perfectly onto their neighboring faces. Split faces, which may be used within local grid refinement, are not considered. Throughout this section, we will mainly focus on the cell-centered finite-volume discretization of the groundwater flow equation, as this expression will be used later on for comparison purposes. Moreover, it is the base for the finite-volume flux reconstruction.

In a cell-centered finite volume method (FVM) applied to the groundwater flow equation (Equation 3), the conservation of fluid mass is enforced via a volume balance, assuming constant water density, for each element E of a given spatial subdivision, or topological triangulation $\mathcal{T}$:

$$\int_E S_0 \, d\mathbf{x} \frac{\partial h}{\partial t} + \int_E \nabla \cdot \mathbf{q} \, d\mathbf{x} = \int_E f \, d\mathbf{x}, \quad \forall E \in \mathcal{T}. \tag{43}$$

For the Richards equation an analogous formulation can be defined using the enlarged storage term (Equation 11) as transient term, and the extended form of Darcy's law (Equation7) yielding a non-linear equation.

Then, the divergence theorem is employed such that the normal component of the specific discharge $\mathbf{n}_E \cdot \mathbf{q}$ is assumed constant on each element face, $F = \partial E_i \cap \partial E_j$, for two neighboring Elements E_i and E_j of a topological triangulation $\mathcal{T}$ leading to conforming fluxes in $\mathcal{RTN}_0$-space:

$$\int_E S_0 \, d\mathbf{x} \frac{\partial h}{\partial t} + \sum_{F \in \mathcal{F}_E} \int_F \mathbf{n}_E \cdot \mathbf{q}_F \, d\mathbf{s} = \int_E f \, d\mathbf{x}, \quad \forall E \in \mathcal{T}, \tag{44}$$

where $\mathbf{q}_F$ is the specific discharge on F (see Figure 3). Because the normal flux $\boldsymbol{\nu}_F \cdot \mathbf{q}_F$ is assumed to be constant on any F, and the source/sink strength f is assumed to be constant on E, the integrals over F, and E reduce to a multiplication with $|F|$, and $|E|$, respectively, assuming an $\mathcal{RTN}_0$-space. Then, in general $[\mathbf{q} \cdot \boldsymbol{\nu}_F]_F = 0$ is assured on any face, while $[h]_F \neq 0$. Higher-order approximations of the specific-discharge on F are possible also in FVM (Brezzi et al., 1985, 1987).

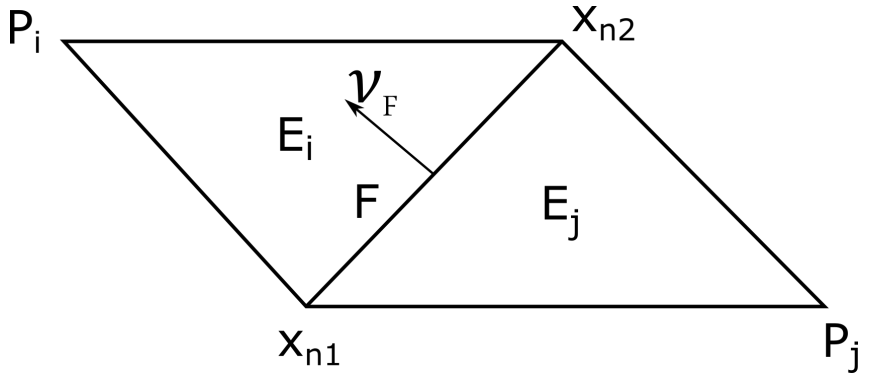

Figure 3: Definitions exemplified for two adjacent triangles. The elements E_i and E_j share a face F on which a unit normal vector $\boldsymbol{\nu}_F$ following a sign convention is defined. The triangular face itself is defined by the line between the two nodes $\mathbf{x}_{n1}$ and $\mathbf{x}_{n2}$. The coordinates of the nodes opposite of F are denoted $\mathbf{P}_i$ for the node in E_i and $\mathbf{P}_j$ for the node in E_j, respectively.(This Figure is taken from Selzer and Cirpka (2020).)

A cell-centered FVM discretization of Equation 3 results in a linear system of equations:

$$\mathbf{S} \frac{\partial \hat{\mathbf{h}}}{\partial t} + \mathbf{A}\hat{\mathbf{h}} = \mathbf{r}, \tag{45}$$

where $\mathbf{S}$ is the specific storage matrix, being a diagonal matrix with the element-wise specific

storage coefficients times the volumes of the elements on the main diagonal for elements not being part of a boundary condition, otherwise the according entry in the main diagonal is a zero. $\hat{\mathbf{h}}$ is the vector of all approximated cell-related hydraulic-head values, $\mathbf{A}$ is the mobility matrix, and $\mathbf{r}$ is a vector containing sources and sinks integrated over the cell-volume, as well as boundary conditions.

In order to construct the mobility matrix $\mathbf{A}$, the normal component of the hydraulic gradient $\boldsymbol{\nu}_F \cdot \nabla h$ at each face needs to be approximated from the head values in the cells. Towards this end, the standard procedure in FVM is to formally allocate the cell-related hydraulic-head value to the cell center, compute the sum of distances of both cell centers to the face, and divide the difference of the cell-related head values by this distance. In our FVM approach, we consider the centroids of triangles, or tetrahedra, as the cell centers rather than the circumcenters because the centroid is in general more representative for an element than the circumcenter, at least if hydraulic conductivity is isotropic. Moreover, the centroid always lies within the respective element, no matter how deformed it is. The disadvantage of taking the centroid as characteristic point of an element in a finite volume formulation is that the line connecting the centroids of two neighboring elements is in general not orthogonal to their shared face, making an orthogonal projection of the distance vector onto the face necessary when computing the associated coefficients of the mobility matrix. Cordes and Kinzelbach (1996) presented a finite volume Method on triangles for isotropic hydraulic conductivity employing circumcenters. Such an approach is easier to compute than employing centroids, as no orthogonal projection is required, but the results are less accurate, and a valid application is restricted to triangles strictly meeting the Delaunay criterion. Otherwise, circumcenters may lie on an element face for right-angled triangles, which could introduce divisions by zero, if the same face is also opposite to a right-angle of the neighboring volume. The circumcenters of elements with angles larger than 90° lie outside of the elements, which can result in coefficients of the mobility matrix with wrong sign.

For an accurate representation of Dirichlet boundaries and Neumann boundaries with a fixed, non-zero flux, we introduce ghost nodes by orthogonally projecting the centroid of an element next to the boundary, $E \in \mathcal{E}_D \cup \mathcal{E}_{N,\pm}$, onto the boundary. That is, we expand the discretization by introducing surface elements of dimension $d-1$. We denote this expanded discretization by $\mathcal{T}_{D,N} = \mathcal{T} \cup \mathcal{F}_D \cup \mathcal{F}_{N,\pm}$, where $\mathcal{F}_D$ is the set of faces discretizing the Dirichlet boundary Γ_D, and $\mathcal{F}_{N,\pm}$ is the union of the sets of faces discretizing the inflow and the outflow boundary, respectively.

The mobility coefficients related to a face of two neighboring elements of the expanded discretization with indices i and j is given by:

$$m_{i,j} = \begin{cases} s_{i,j}^{\perp} \frac{|F|}{\frac{l_i}{k_i} + \frac{l_j}{k_j}}, & F = \partial E_i \cap \partial E_j,\ E_i, E_j \in \mathcal{T}, \\ \frac{|F| k_i}{l_i}, & F = \partial E_i \cap \Gamma_D,\ E_i \in \mathcal{E}_D, \\ c, & E_i \in \mathcal{F}_{N,\pm}, \end{cases} \tag{46}$$

where k is the element-wise isotropic hydraulic conductivity, and $c > 0$ is a scalar. Note that the discretization of the fluxes according to Equation 65 is based on a two-point flux approximation, therefore, the introduction of anisotropy in grids which are not aligned with the hydraulic conductivity tensor, i.e, k-orthogonal, is not possible without accuracy loss. Finite volume methods employing full material tensors have been developed for node-centered dual grids based on a primal grid of triangles or deformed quadrilaterals (e.g., Edwards (2002)), and for general grids in two dimensions including cell-centered triangular grids (Friis et al., 2009). All these methods include transformations of coordinates, which are beyond the scope of our present consideration. Another possible route of including anisotropy would be following up on the approach of Su et al. (2020) using orthogonal projections and a multi-point flux approximation.

For adapting Equation 65 in accordance with Richards' equation, one would add relative permeability as another multiplicative factor in the first two sub-cases of Equation 65. Here, one would apply upstream weighting in analogy to Equation 67, between the centroids of two elements i and j. Note that relative permeability is a property of the cell-center being a non-linear function of the pressure head at the cell-center in a finite volume formulation.

Furthermore, if ℓ is the distance between two centroids, ℓ_i is the part of ℓ in the element i, and ℓ_j is the part of ℓ in the neighboring element j, and $s^{\perp}_{i,j}$ is a scaling factor such that only the orthogonal part of a mobility coefficient normal to an internal face is considered:

$$s^{\perp}_{i,j} = \frac{\|\boldsymbol{\ell} - \frac{\langle \mathbf{f}, \boldsymbol{\ell} \rangle}{\langle \mathbf{f}, \mathbf{f} \rangle} \mathbf{f}\|_2}{\|\boldsymbol{\ell}\|_2}, \qquad F = \partial E_i \cap \partial E_j,\ E_i, E_j \in \mathcal{T}, \tag{47}$$

where $\boldsymbol{\ell}$ is the distance vector between the centroids of two neighboring elements E_i and E_j, and $\mathbf{f}$ is the vector lying on $F = \partial E_i \cap \partial E_j$. For triangles and other two-dimensional elements, the evaluation of Equation 47 is straight-forward. For some grids or neighboring elements in $\mathbb{R}^3$, Equation 47 is topological two-dimensional, too. In any case, $\mathbf{f}$ can be the tangential part of $\boldsymbol{\ell}$ on F, eventually being scaled, such that the remaining part of $\boldsymbol{\ell}$ is normal to F. However, reformulating Equation 47 facilitates a more general, and much easier numerical evaluation. Therefore, we redefine the orthogonal projection 47 to be a factor employing just the normal component of the direct connection of the cell centers of E_i and E_j on the face $F = \partial E_i \cap \partial E_j$. Then Equation 47 can be rewritten as:

$$s^{\perp}_{i,j} = \frac{\ell^{\perp}}{\ell} = \frac{\|\boldsymbol{\nu}_F \cdot \boldsymbol{\ell}\|_2}{\|\boldsymbol{\ell}\|_2}, \qquad F = \partial E_i \cap \partial E_j,\ E_i, E_j \in \mathcal{T}. \tag{48}$$

Then, the element A_{ik} of the mobility matrix $\mathbf{A} \in \mathbb{R}^{N \times N}$ with two indices i and j ranging over

the elements in $\mathcal{T}_{D,N}$ is given by:

$$A_{ik} = \begin{cases} \sum\limits_{j:E_i \cap E_j = F} m_{i,j}, & i = k,\ E_i \in \mathcal{T}, \\ 1, & i = k,\ E_i \in \mathcal{F}_D, \\ m_{i,k}, & i = k,\ E_i \in \mathcal{F}_{N,\pm}, \\ -m_{i,k}, & i \neq k,\ E_i \cap E_j = F, \\ 0, & \text{otherwise}, \end{cases} \tag{49}$$

and the right-hand side vector $\mathbf{r} \in \mathbb{R}^N$ is given by:

$$r_i = \begin{cases} f_i |E_i|, & \text{if } E_i \in \mathcal{T}, \\ -\mathbf{n} \cdot \mathbf{q}_{fix,i} |F_i|, & \text{if } F_i \in \mathcal{F}_{N,\pm}, \\ \hat{h}_{i,D}, & \text{if } E_i \in \mathcal{F}_D, \end{cases} \tag{50}$$

where f_i is the volumetric source/sink strength per unit volume for an element E_i, $\hat{h}_{i,D}$ is the assigned Dirichlet boundary condition on the Dirichlet face $E_i \in \mathcal{F}_D$, and $\mathbf{q}_{fix,i}$ is the specific discharge on a Neumann face with non-zero flux, $E_i = F_i \in \mathcal{F}_{N,\pm}$.

2.5 Postprocessing Methods for Non-Conforming Velocity Fields

In this section two different flux postprocessing techniques are presented. The first is the $\mathcal{RTN}_0$-projection which can be formulated fully consistently for linear problems like saturated one-phase flow on simplicial grids. A standard finite-element solution on simplices yields a unique gradient on the elements, and therefore, an element-wise constant velocity vector. Such a velocity field can be easier projected onto a conforming one, than a velocity field which varies within the elements, which would require at best a multipoint projection and transformations of coordinates for deformed non-simplicial elements to be most accurate. However, an application to deformed non-simplicial elements is possible and an extension to variably saturated flow is outlined, too. The $\mathcal{RTN}_0$-projection and its description has been published in Selzer and Cirpka (2020).

The second postprocessing technique is not a projection of a primal velocity field, but a flux reconstruction based on lowest-order finite volumes, and a two-point flux approximation scheme. The big advantage of this technique is the high stability of this scheme, and its computational speed. Moreover, it can easily be applied to non-linear equations like the Richards equation without the possible introduction of severe numerical artifacts. The finite-volume flux reconstruction and its description has been published in Selzer et al. (2021).

2.5.1 The $\mathcal{RTN}_0$-Projection

The $\mathcal{RTN}_0$-projection aims to project a non-conforming, non-mass-conservative velocity field onto a mass-conservative and conforming one. This is done by minimizing the element-wise

residual between the hydraulic gradient of the $\mathcal{P}_1$ Galerkin FEM solution for Darcy flow and a solution corresponding to the velocity approximation in $\mathcal{RTN}_0$-space, in which the normal component of the velocity is constant on and continuous across every face, and the velocity components vary linearly within each element.

The $\mathcal{RTN}_0$ basis functions, $\boldsymbol{\psi}_F(\boldsymbol{x})$, can be set globally for the topological triangulation $\mathcal{T}$ for all elements sharing face F. Any face is either shared by two neighboring elements, $F = \partial E_i \cap \partial E_j$, or is a face of only one element, $F \in \mathcal{F}_D \cup \mathcal{F}_N$. For each face $F = \partial E_i \cap \partial E_j$ there is one vertex in both elements E_i and E_j each that is opposite of the face F, or F is a face of only one element, $F \subset \partial E_i$. $\mathbf{P}_i$ denotes the coordinates of the node opposite to F in element E_i, and $\mathbf{P}_j$ those of the node opposite to F in E_j, if there is an element E_j. Then the basis function $\boldsymbol{\psi}_F(\mathbf{x})$ is:

$$\boldsymbol{\psi}_F(\mathbf{x}) = \begin{cases} \mathbf{n}_{E_i} \cdot \boldsymbol{\nu}_F \frac{|F|}{d|E_i|} \left(\mathbf{x} - \mathbf{P}_i\right), & \mathbf{x} \in E_i, \\ \mathbf{n}_{E_j} \cdot \boldsymbol{\nu}_F \frac{|F|}{d|E_j|} \left(\mathbf{x} - \mathbf{P}_j\right), & \mathbf{x} \in E_j, \\ 0, & \text{elsewhere.} \end{cases} \tag{51}$$

Then the element-wise $\mathcal{RTN}_0$ velocity field in global coordinates for a simplicial element E_i is given by:

$$\mathbf{q}_{E_i}^{RTN_0}(\mathbf{x}) = \left(a_j^q + b^q x_j\right)_{j=1,\ldots,d} = \sum_{F \in \mathcal{F}_{E_i}} \boldsymbol{\psi}_F(\mathbf{x}) q_F^{\perp}, \quad \mathbf{x} \in E_i, \tag{52}$$

where $q_F^{\perp} = \boldsymbol{\nu}_F \cdot \mathbf{q}_F$ is the orthogonal component of the specific-discharge vector on F, which is assumed to be constant on the face F, a_j^q is an individual scalar for each spatial direction x_j, b^q is a scalar independent of the spatial direction equaling the divergence of the velocity field integrated over E_i times $(d|E_i|)^{-1}$, therefore, $b^q = 0$ holds for non-divergent flow. It should be noted that also in cell-centered FVM fluxes normal to the faces, $q_F^{\perp}$, are directly computable, such that the $\mathcal{RTN}_0$ basis functions of Equation 52 can be used to obtain the velocity field within the cells of an FVM. For a divergent velocity field, the $\mathcal{RTN}_0$ velocity approximation varies within the element, and the same holds for an FVM velocity field with internal sources or sinks. For illustration, see Figure 4 showing a divergent and a non-divergent velocity field. To make specific-values comparable on an element by element basis, we evaluate the velocity at the element centroid, and denote such an element-wise velocity originating from the FVM discretization $\mathbf{q}_{E_i}^{FVM}(\mathbf{x}_c)$.

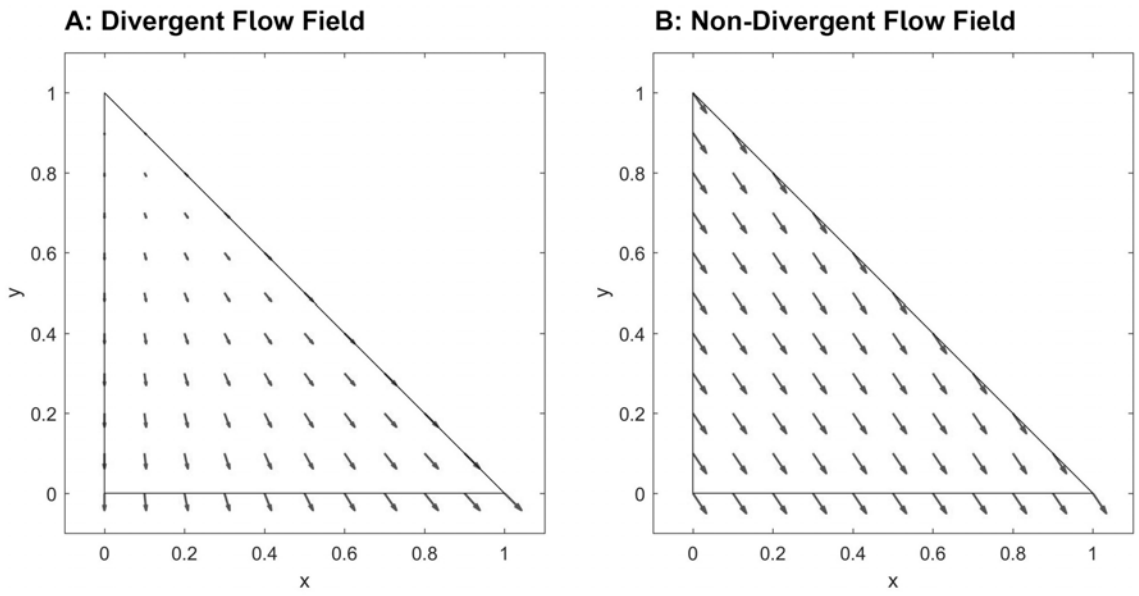

Figure 4: $\mathcal{RTN}_0$ velocity fields. a) divergent flow field, in which flow leaves only at the bottom face. The resulting velocity varies linearly in every spatial dimension. This velocity field is equivalent to the velocity base function $\boldsymbol{\psi}_F(\mathbf{x})$ related to the normal flux density $q_F^{\perp}$ at the bottom face. b) non-divergent, uniform flow field, resulting from the superposition of three base functions in the case that the total flux across all faces equals zero. (This Figure is taken from Selzer and Cirpka (2020).)

If $\mathbf{q}_{E_i}^{RTN_0}$ is evaluated at the centroid $\mathbf{x}_c$ of an element E_i, an element-wise residual vector $\boldsymbol{\epsilon}_{q,E_i}$ between the $\mathcal{RTN}_0$-based velocity $\mathbf{q}_{E_i}^{RTN_0}$ and the element-wise constant $\mathcal{P}_1$-Galerkin-FEM velocity $\mathbf{q}_{E_i}^{FE}$ can be determined:

$$\boldsymbol{\epsilon}_{q,E_i} = \mathbf{q}_{E_i}^{RTN_0} - \mathbf{q}_{E_i}^{FE}. \tag{53}$$

However, it is advantageous to consider the residual of the hydraulic gradient at the centroid instead of the specific discharge within the proposed global optimization procedure. If the specific discharge was considered directly, residuals in zones of higher hydraulic conductivity, which are small in relative terms, would have a higher contribution to the objective function of the optimization procedure than residuals in zones of lower hydraulic conductivity, which are high in relative terms. We propose to minimize the $\mathcal{L}^2$-norm of the hydraulic gradient at the centroids $\mathbf{x}_c$, which is a slight adaption of the approach of Odsæter et al. (2017), who considered normal components of hydraulic gradients at the faces. Our residual vector at the centroid of the single element E_i is:

$$\begin{aligned}\boldsymbol{\epsilon}_{\nabla h,E_i} &= -\mathbf{K}_{E_i}^{-1}\mathbf{q}_{E_i}^{RTN_0}(\mathbf{x}_c) - \nabla\boldsymbol{\chi}_{E_i}\hat{\mathbf{h}} \\ &= -\mathbf{K}_{E_i}^{-1}\sum_{F\in\mathcal{F}_{E_i}}\boldsymbol{\psi}_F(\mathbf{x}_c)q_F^{\perp} - \nabla\boldsymbol{\chi}_{E_i}\hat{\mathbf{h}},\end{aligned} \tag{54}$$

where $\nabla\boldsymbol{\chi}_{E_i}$ is the gradient of the linear shape functions $\boldsymbol{\chi}_{E_i}$ in element E_i, and $\mathbf{K}_{E_i}$ is the hydraulic-conductivity tensor associated to element E_i. In a finite-element-type primal solu-

tion of Richards' equation we replace $\mathbf{K}_{E_i}^{-1}$ by the inverse of the variably saturated hydraulic-conductivity tensor, $k_{rel,c_i}\mathbf{K}_{E_i}$, where k_{rel,c_i} is the relative permeability associated to the centroid, c_i, of element E_i. However, in a finite-element-type discretization, the relative permeability is a nodal property and a function of the pressure head. Moreover, several relative permeability values may be associated to the nodes, depending on which element sharing the node is considered, if unsaturated material properties differ in the adjacent elements and flow is unsaturated. For an extension to variably saturated flow, we therefore need to interpolate the relative permeability to the element centroids. We do this in our application by linear interpolation of the nodal relative permeabilities to the centroid for each element, this procedure is equal to taking the arithmetic mean of the nodal relative permeabilities.

Considering Equation 54, a global system of equations can be set up for the squared element-wise errors:

$$\begin{aligned}\langle \boldsymbol{\epsilon}_{\mathcal{T}}, \boldsymbol{\epsilon}_{\mathcal{T}} \rangle =& \boldsymbol{\epsilon}_{\mathcal{T}}^T \boldsymbol{\epsilon}_{\mathcal{T}} \\ =& (q_{F_m}^{\perp})_{F_m \in \mathcal{F}}^T \mathbf{N}^T \mathbf{N} (q_{F_m}^{\perp})_{F_m \in \mathcal{F}} \\ & - 2((\nabla h)_{E_i}^{FE})_{E_i \in \mathcal{T}} \mathbf{N} (q_{F_m}^{\perp})_{F_m \in \mathcal{F}} \\ & + ((\nabla h)_{E_i}^{FE})_{E_i \in \mathcal{T}}^T ((\nabla h)_{E_i}^{FE})_{E_i \in \mathcal{T}},\end{aligned} \tag{55}$$

where the index i ranges over the elements, and m is an index ranging over the faces, both according to a global numbering convention. Furthermore, $\boldsymbol{\epsilon}_{\mathcal{T}} \in \mathbb{R}^{Nd}$ is the vector of element-wise errors in each spatial direction, where Nd is the number of elements times the number of dimensions, $(q_{F_m}^{\perp})_{F_m \in \mathcal{F}} \in \mathbb{R}^M$ is the vector of normal components of the specific discharge following global indexing of faces, where no-flow boundaries are excluded. In the following, $((\nabla h)_{E_i}^{FE})_{E_i \in \mathcal{T}} \in \mathbb{R}^{Nd}$ is the vector of element-wise hydraulic gradients with $(\nabla h)_{E_i}^{FE} = \nabla \boldsymbol{\chi}_{E_i} \hat{\mathbf{h}}$ obtained by a $\mathcal{P}_1$ Galerkin FEM solution, and $\mathbf{N} \in \mathbb{R}^{Nd \times M}$ is the matrix, in which M is the number of faces minus the number of faces belonging to no-flow boundaries, containing all $\mathcal{RTN}_0$ basis functions $\boldsymbol{\psi}_F(\mathbf{x}_c)$, evaluated at the centroids, scaled by $\mathbf{K}_{E_i}^{-1}$, or alternatively by $(k_{rel,c_i}\mathbf{K}_{E_i})^{-1}$ for a finite-element-type primal solution of Richards' equation, according to a global numbering scheme of elements and faces:

$$\mathbf{N}_{im} = \begin{cases} \mathbf{K}_{E_i}^{-1} \boldsymbol{\psi}_{F_m}(\mathbf{x}_c), & \text{if } E_i \in \mathcal{T},\ F_m \subset \partial E_i,\ F_m \in \mathcal{F}_I \cup \mathcal{F}_D, \\ 0, & \text{otherwise,} \end{cases} \tag{56}$$

where $\boldsymbol{\psi}_{F_m}(\mathbf{x}_c)$ is the $\mathcal{RTN}_0$ basis function of face F_m evaluated at the respective component of the centroid. $\mathbf{N}_{im}$ has d elements, so that $\mathbf{N}_{im} q_{F_m}^{\perp}$ are the contributions of the normal flux at face F_m to the hydraulic-gradient vector at the centroid of element E_i. The full matrix $\mathbf{N} \in \mathbb{R}^{Nd \times M}$ is assembled from the individual contributions $\mathbf{N}_{im}$.

The squared errors of Equation 55 are minimized under the constraint that mass is conserved

element-wise, leading to the following auxiliary condition:

$$\mathbf{M}(q^{\perp}_{F_m})_{F_m\in\mathcal{F}} = \left(\int\limits_{E_i} f_{E_i} d\mathbf{x}\right)_{E_i\in\mathcal{T}}, \tag{57}$$

where i ranges over all elements and m ranges over all faces following a global numbering scheme, and f_{E_i} is the divergence of the specific discharge in element E_i which equals the source/sink strength. For transient simulations, the change in storage could be lumped in f_{E_i}, too. By this, we would would treat the change in storage as an additional source/sink term, which is possible in a postprocessing scheme. $\mathbf{M} \in \mathbb{R}^{N\times M}$ is a matrix with the following elements:

$$M_{im} = \begin{cases} \boldsymbol{\nu}_{F_m} \cdot \mathbf{n}_{E_i} |F_m|, & \text{if } E_i \in \mathcal{T},\ F_m \subset \partial E_i,\ F_m \in \mathcal{F}_I \cup \mathcal{F}_D, \\ 0, & \text{otherwise.} \end{cases} \tag{58}$$

Minimizing the squared residuals of Equation 55 subject to the constraint of Equation 57 is done by the method of Lagrange multipliers, in which the following objective function is minimized:

$$\begin{aligned} L((q^{\perp}_{F_m})_{F_m\in\mathcal{F}}, \boldsymbol{\lambda}) =& \frac{1}{2}\boldsymbol{\epsilon}_{\mathcal{T}}^T \boldsymbol{\epsilon}_{\mathcal{T}} \\ &+ \boldsymbol{\lambda}\left(\mathbf{M}(q^{\perp}_{F_m})_{F_m\in\mathcal{F}} - (|E_i| f_{E_i})_{E_i\in\mathcal{T}}\right), \end{aligned} \tag{59}$$

where $\boldsymbol{\lambda} \in \mathbb{R}^N$ is the vector of element-wise Lagrange multipliers. To obtain the minimum of the objective function, the derivatives with respect to $(q^{\perp}_{F_m})_{F_m\in\mathcal{F}}$ and $\boldsymbol{\lambda}$ must be zero:

$$\begin{aligned} \frac{\partial L}{\partial (q^{\perp}_{F_m})_{F_m\in\mathcal{F}}} &= \mathbf{N}^T\mathbf{N}(q^{\perp}_{F_m})_{F_m\in\mathcal{F}} \\ &\quad -\mathbf{N}^T((\nabla h)^{FE}_{E_i})_{E_i\in\mathcal{T}} + \mathbf{M}^T\boldsymbol{\lambda}^T \overset{!}{=} \mathbf{0}, \end{aligned} \tag{60}$$

$$\frac{\partial L}{\partial \boldsymbol{\lambda}} = \mathbf{M}(q^{\perp}_{F_m})_{F_m\in\mathcal{F}} - (|E_i| f_{E_i})_{E_i\in\mathcal{T}} \overset{!}{=} \mathbf{0}, \tag{61}$$

leading to the following system of equations in block matrix form:

$$\begin{bmatrix} \mathbf{N}^T\mathbf{N} & \mathbf{M}^T \\ \mathbf{M} & \mathbf{0} \end{bmatrix} \begin{bmatrix} (q^{\perp}_{F_m})_{F_m\in\mathcal{F}} \\ \boldsymbol{\lambda}^T \end{bmatrix} = \begin{bmatrix} \mathbf{N}^T((\nabla h)^{FE}_{E_i})_{E_i\in\mathcal{T}} \\ (|E_i| f_{E_i})_{E_i\in\mathcal{T}} \end{bmatrix}. \tag{62}$$

Equation 62 describes the system of equations of the $\mathcal{RTN}_0$-projection, which is square and has the order $M+N$ independent of the spatial dimensionality d. For incorporating a fixed, non-zero flux boundary condition with a normal flux component $q^{\perp}_{F_m}$ we may replace the corresponding entry on the main diagonal of the matrix by a one, set all other entries of the respective row to zero, and specify the normal component of the flux in the entry m of the right-hand side vector. Alternatively, we may not consider the according face F_m explicitly, neglect it in the equation system (Equation 62) to solve, and specify an integrated source/sink contribution to add, which is equal to the normal flux component $q^{\perp}_{F_m}$ integrated over the face F_m for the

element E_i featuring the fixed flux boundary condition under consideration. In a postprocessing step, the normal flux component $q^{\perp}_{F_m}$, being a Neumann boundary condition, is then mapped to the according face F_m to meet the boundary condition and to achieve element-wise mass conservation.

2.5.2 The Finite-Volume Flux Reconstruction

We assume that a node-centered solution of Equation 6 is already available, typically computed by a finite element method or by some finite difference scheme. Such a solution is known to be not mass-conservative in the elements in general (e.g., Cordes and Kinzelbach, 1992; Larson and Niklasson, 2004; Odsæter et al., 2017; Putti and Sartoretto, 2009). In finite element models, the normal velocity component exhibits a jump across element faces (e.g., Odsæter et al., 2017), leading to an inconsistent velocity field, which is not suitable for particle tracking. In standard finite difference methods, the velocity component normal to a cell boundary is not defined. While cell-centered finite volume schemes are sometimes denoted finite difference methods by their developers (e.g., McDonald and Harbaugh, 1988), the original finite difference method does not perform a mass balance in cells, and mass conservation is not guaranteed.

We use HydroGeoSphere in finite-difference mode to obtain a steady-state variably saturated flow solution (Aquanty, Inc., 2015; Therrien and Sudicky, 1996). The finite difference method of HydroGeoSphere is formulated in finite-element terms that neglects the cross terms, so that only direct connections between nodes that correspond to the primal grid are retained (Therrien and Sudicky, 1996). Thus, the approximation of the hydraulic-head field may be imagined as a continuous solution on these direct connections resembling a network of lines on element edges. When visualizing the hydraulic head solution, a bilinear interpolation within the prismatic elements is typically assumed, which is not fully consistent with the way heads have been computed.

Taking the HydroGeoSphere pressure-head solution to obtain nodal values of the relative permeability in each element and the storage-related divergence of the flow field, we present a flux reconstruction that is locally mass-conservative in the elements and conforming on the element faces. To do so, we recompute the hydraulic head governed by the Richards equation (Equation 6) with a lowest-order, cell-centered finite volume method using the elements as control volumes. From the finite-volume solution we can construct the velocity field in lowest-order Raviart-Thomas-Nédélec ($\mathcal{RTN}_0$) space on triangular prisms, forming the basis for consistent particle tracking. The following description of the lowest-order finite volume flux reconstruction heavily relies on the description given by Selzer and Cirpka (2020). This section rephrases some equations of section 2.4.3 where the basis of a lowest-order finite volume method for unstructured grids is discussed. This section puts the concepts and equations of section 2.4.3 into the context of a flux reconstruction, if a primal solution for the hydraulic heads already exists. For this purpose some equations are repeated here, though sometimes in a slightly different notation to account for the specifics of a flux reconstruction compared to a numerical scheme designed for computing a primal solution.

Finite volume schemes are based on conserving the quantity of interest, in our case the fluid mass assuming constant fluid density, in each element:

$$\int_E \nabla \cdot \mathbf{q}\, d\mathbf{x} = \int_E f_*\, d\mathbf{x}, \quad \forall E \in \mathcal{T}, \tag{63}$$

where $f_* = f - \partial\theta/\partial t$ includes the expanded change in storage (Equation 11). Applying the divergence theorem yields:

$$\sum_{F \in \mathcal{F}_E} \int_F \mathbf{n}_{E,F} \cdot \mathbf{q}\, d\mathbf{s} = \int_E f_*\, d\mathbf{x}, \quad \forall E \in \mathcal{T}. \tag{64}$$

For setting up the system of linear equations, we expand the discretization by treating Dirichlet and Neumann faces as ghost elements, implying explicit head values at these faces. A lowest-order finite difference approximation of the flux on the boundary of the elements together with a weighted approximation of the material coefficients yields the mobility coefficients, $m_{i,j}$, on the faces for neighboring elements:

$$m_{i,j} = \begin{cases} s_{i,j}^{\perp} k_r^{i,j} |F| \frac{\ell_i K_i + \ell_j K_j}{\ell^2}, & \text{on } F = \partial E_i \cap \partial E_j \in \mathcal{F}_I, \\ \frac{|F| k_r^{i,j} K_i}{\ell_i}, & \text{on } F = \partial E_i \cap \Gamma_D \in \mathcal{F}_D, E_j \subset \Gamma_D, \\ c, & \text{on } F \in \mathcal{F}_N, \end{cases} \tag{65}$$

where i and j are the indices of two neighboring elements, and K_i and K_j are the isotropic, saturated hydraulic conductivities of the elements i and j, respectively. In the anisotropic case, these saturated conductivity values would represent the conductivity in the direction of the normal vector on face F. Furthermore, $\ell = \ell_i + \ell_j$ is the distance between the centroids of the triangular prisms i and j, whereas ℓ_i and ℓ_j are the parts within elements i and j, respectively. $s_{i,j}^{\perp}$ is an orthogonal correction factor accounting for the fact that the approximated flux is in general not normal to the face $F = \partial E_i \cap \partial E_j$. $k_r^{i,j}$ is the relative permeability on the face F, and $c > 0$.

Including anisotropy as a full tensor would be possible in a cell-centered finite volume flux reconstruction, but it is beyond our consideration. As an example, Edwards (2002) derived such techniques for a dual grid to unstructured triangular meshes, Friis et al. (2009) and Edwards and Zheng (2010) extended this approach to the primal grid using a dual grid and transformations of coordinates similar to those known from finite elements on subspaces, which yields a more versatile finite-volume formulation, but also adds to the computational costs. Another possible route for including anisotropy may be facilitated by incorporating a flux approximation scheme similar to the one suggested by Su et al. (2020).

As mentioned above, the discretization $\mathcal{T}$ is expanded by ghost elements on faces in order to account for Dirichlet and Neumann boundary conditions not being no-flow boundary conditions.

In contrast to a standard cell-centered finite volume discretization, we do not use the distance-weighted harmonic mean to compute the mobility coefficients but a distance-weighted arithmetic mean. This turned out to be more stable and better mimics the primal hydraulic head solution obtained by the finite-difference mode of HydroGeoSphere. Celia et al. (1990) also used an (unweighted) arithmetic mean of hydraulic conductivities to obtain an equivalent approximation of spatial derivatives between a finite difference method and the standard Galerkin finite element method on a one-dimensional regular grid.

In order to compute the correction factor $s_{i,j}^{\perp}$ we repeat Equation 48 and consider the projection of the distance vector $\boldsymbol{\ell}$, connecting the centroids of the elements i and j, onto the direction of the normal vector on a face F:

$$s_{i,j}^{\perp} = \frac{\ell^{\perp}}{\ell} = \frac{\|\boldsymbol{\nu}_F \cdot \boldsymbol{\ell}\|_2}{\|\boldsymbol{\ell}\|_2}, \qquad F = \partial E_i \cap \partial E_j,\ E_i, E_j \in \mathcal{T}. \tag{66}$$

For the approximation of the relative permeability on the faces, $k_r^{i,j}$, we need to consider that the finite-element-type scheme may yield different values of k_r for the same node in the different elements i and j sharing the node, if material properties differ between elements i and j. For a flux reconstruction, we need to approximate a single relative permeability per face. We do this by first linearly interpolating the nodal values belonging to each element to the centroid of the face, which is the arithmetic mean of the nodal values, and then applying upstream weighting, that is taking the relative permeability of the element that has the larger hydraulic head $h = z + \psi$ at its element centroid in the initial finite-element or finite-difference solution:

$$k_r^{i,j} = \begin{cases} \frac{1}{|\mathcal{N}_F|} \sum\limits_{n \in \mathcal{N}_F} k_r^{i,n}, & F \in \mathcal{F}_{E_i}, \quad \text{if} \quad h_{c,i} \geq h_{c,j} \quad \text{or} \quad F \subset \Gamma, \\ \frac{1}{|\mathcal{N}_F|} \sum\limits_{n \in \mathcal{N}_F} k_r^{j,n}, & F \in \mathcal{F}_{E_j}, \quad \text{if} \quad h_{c,j} > h_{c,i}, \end{cases} \tag{67}$$

where $|\mathcal{N}_F|$ is the number of nodes on face F, $k_r^{i,n}$ and $k_r^{j,n}$ are the relative permeabilities at node n in the elements i and j, respectively, whereas $h_{c,i}$ and $h_{c,j}$ are the hydraulic heads at the centroids of elements i and j, respectively. Additionally, we set the relative permeability of a node exhibiting recharge to one, which we found necessary for accurately mimicking some primary solutions. As these nodes experience a water influx during a recharge event, the water saturation is close to unity anyway.

With these approximations, we can specify the resulting system of linear equations by assembling the mass-balance equations for all elements:

$$\mathbf{A}\hat{\mathbf{h}} = \mathbf{r}, \tag{68}$$

where $\mathbf{A}$ is the mobility matrix, $\hat{\mathbf{h}}$ is a pseudo hydraulic head, which we call the pseudo-potential, defined on the centroids of the elements, which leads to mass-conservative and conforming fluxes, and $\mathbf{r}$ is the right-hand side vector including the divergence of the flux f_* integrated

over the element, as well as boundary conditions. After computing the pseudo-potential, we can reconstruct the mass-conservative, lowest-order fluxes on the faces which are a prerequisite for particle tracking in $\mathcal{RTN}_0$-space.

In the following, we give the mass-balance equation for an inner element E_i:

$$\text{for } E_i \in \mathcal{T} : \sum_{j:E_i \cap E_j = F} m_{i,j}(h_i - h_j) = f_{*,i}|E_i|, \tag{69}$$

where h_i is the pseudo-potential at the centroid of element i, whereas h_j is either the corresponding pseudo-potential at the centroid of element j or the fixed potential at a Dirichlet boundary condition $h_{D,m}$, or the pseudo-potential associated to a Neumann boundary condition $h_{N,m}$, which is set such that the desired flux between the Neumann boundary and the i-th element is obtained. Furthermore, $f_{*,i}$ is the source or sink strength including the change in storage associated to the centroid of the element E_i. A Robin boundary condition is applied by setting a reference potential $h_{R,m}$, and giving a conductance C_m associated to the face m for computing the total flux towards the i-th element. For setting a Robin boundary condition Equation 69 is superimposed with:

$$C_m(h_i - h_{R,m}) = Q_m, \tag{70}$$

where Q_m is a total normal flux on face m, which leads to an additional mobility coefficient on the main diagonal of matrix $\mathbf{A}$ in Equation 68 and an entry in the right-hand side vector of the system 68.

If element i is a face belonging to a Dirichlet or a Neumann boundary, the resulting i-th equation is:

$$\text{for } E_i = F_m \in \mathcal{F}_D : \qquad h_i = h_{D,m}, \tag{71}$$

$$\text{for } E_i = F_m \in \mathcal{F}_N : \qquad c_m(h_{N,m} - h_j) = q_{N,m} \cdot |F_m|, \tag{72}$$

where $h_{D,m}$ is the hydraulic head associated to a Dirichlet face, $h_{N,m}$ is a hydraulic head at a Neumann face defined such that the normal component of the specific discharge, entering or exiting through element E_i, equals $q_{N,m}$ including a sign denoting the direction of the flux, and c_m is a positive scalar.

2.6 Semi-Analytical Particle Tracking in $\mathcal{RTN}_0$-Space

In advective particle tracking, the trajectories of ideal, point-like particles are computed via integrating the particle velocity over the travel time. A trajectory $\mathbf{X}_p(t) = (x_{j,p}(t))_{j=1,\dots,d}$ in $\mathbf{x} \in \Omega$ for a particle p is given by:

$$\mathbf{X}_p(t_0 + \Delta t) = \mathbf{X}_p(t_0) + \int_{t_0}^{t_0+\Delta t} \mathbf{v}(\mathbf{X}_p(\tau), \tau)\, d\tau, \tag{73}$$

where t_0 is the starting time, Δt is a discrete time increment, $\mathbf{v}$ is a velocity field and τ is the travel time. Note that for us the velocity, $\boldsymbol{v}$, is the linear average velocity being $\mathbf{v} = \mathbf{q}/\theta$, where θ is the volumetric water content, which reduces to the flow-effective porosity n_e in saturated flow. Both, the volumetric water content as well as the flow-effective porosity are assumed to be constant in the elements. If volumetric water contents are given on the nodes within a primal hydraulic head/pressure solution, the element-wise constant volumetric water content is given by linear interpolation to the element centroid, which equals the arithmetic mean of the nodal values. The flow-effective porosity is a material property defined constant on the elements anyways.

In the following, we present element-wise analytical solutions of Equation 73 to compute the exit location of a particle on an element-face from the start location of that particle within the element, or from the entry location on another element-face, for non-divergent and divergent flow on arbitrarily shaped triangles and tetrahedra in global coordinates. Note that, if there is an internal sink or a stagnation point in the element, and the analytical solution is that the particle will stop within the element, this analytical solution is recovered, too, by the given analytical solutions. Here, the trajectory is followed towards the face, where the particle should head towards but stopped at the center of the sink, or the stagnation point. For triangular prisms, which is our example of a superposition of two simplices, we first give the according $\mathcal{RTN}_0$-space on triangular prisms. Based on this, we derive a particle tracking scheme in a mixed coordinate system, combining particle tracking in global, physical coordinates in the horizontal with tracking in the vertical in a local coordinate system. With this, we can track a particle from one element-face to the next throughout the whole domain until a boundary or an internal sink like an extraction well. The analytical solution for triangles and tetrahedra are described by Selzer and Cirpka (2020), while the $\mathcal{RTN}_0$-space in mixed coordinates for triangular prisms and the associated particle tracking scheme is described by Selzer et al. (2021).

2.6.1 Analytical Solutions for Triangles

Particle tracking in non-divergent flows on triangles We first consider non-divergent flow on triangles, which implies an element-wise constant velocity vector. Given an entry point of a particle on the starting face F_s, the element-wise constant velocity vector defines a straight line, starting at the entry point. The point of the first intersection of this line with one of the element faces not being the starting face is the exit point. It is computed by intersection of hyperplanes, i.e., lines ($d = 2$), or planes ($d = 3$), containing the element faces with the trajectory line. Let $\mathbf{x}_p = (x_p, y_p)$ be the starting point of a particle in the element E_i, or on the face $F_s \in \mathcal{F}_{E_i}$. Then, the other points lying on the trajectory, given the velocity vector, $(v_{j,E_i})_{j=1,\dots,d} = \mathbf{v}_{E_i}$, starting from $\mathbf{x}_p$, is given by $\mathbf{x}_l = (x_l, y_l) = \mathbf{x}_p + r\mathbf{v}_{E_i}$, where $r > 0$ is a positive scalar. Every face F of an element E_i can be described by two nodes $\mathbf{x}_{n1} = (x_{n1}, y_{n1})$ and $\mathbf{x}_{n2} = (x_{n2}, y_{n2})$.

Then, a possible exit point, $\mathbf{x}_{e,p} = (x_{e,p}, y_{e,p})$, on the face F is given by line-line intersection:

$$\mathbf{x}_{e,p} = \left(\frac{(x_{n2} - x_{n1}) \cdot (x_l \cdot y_p - x_p \cdot y_l) - (x_l - x_p) \cdot (x_{n2} \cdot y_{n1} - x_{n1} \cdot y_{n2})}{(y_{n2} - y_{n1}) \cdot (x_l - x_p) - (y_l - y_p) \cdot (x_{n2} - x_{n1})}, \right.$$
$$\left. \frac{(y_p - y_l) \cdot (x_{n2} \cdot y_{n1} - x_{n1} \cdot y_{n2}) - (y_{n1} - y_{n2}) \cdot (x_l \cdot y_p - x_p \cdot y_l)}{(y_{n2} - y_{n1}) \cdot (x_l - x_p) - (y_l - y_p) \cdot (x_{n2} - x_{n1})} \right) \quad (74)$$
$$\forall F \in \mathcal{F}_{E_i} \setminus \{F_s\},$$

where F_s is the face on which the particle starts, if the particle starts on a face. Let $\mathbf{x}_d$ be the distance vector $\mathbf{x}_{e,p} - \mathbf{x}_p$, then the travel times leading to possible intersection points are simply given by:

$$\boldsymbol{\tau} = \left(\frac{||\mathbf{x}_d||_2}{||\mathbf{v}_{E_i}||_2} \right)_{F \in \mathcal{F}_{E_i} \setminus \{F_s\}} \text{with} \operatorname{sign}(x_j) = \operatorname{sign}(v_{j,E_i}), \quad (75)$$

The actual exit point $\mathbf{x}_e$ is the possible exit point with the corresponding smallest positive travel time:

$$\mathbf{x}_e = (\mathbf{x}_{e,p})_i \text{ with } i = \min\{k : \tau_k = \min_{\substack{1 \le j \le n, \\ \tau_j > 0}} \tau_j\}, \quad (76)$$

where i is the corresponding index of the possible exit point with the corresponding minimum travel time τ_i.

Particle tracking in divergent flows on triangles For divergent flow on triangles in global coordinates, we again consider the intersection of the particle trajectory within an element with the element faces. Considering the space defined in Equation 18, and Equations 51, 52, we know that for every spatial dimension x_j with $\mathbf{x} = (x, y)$, the velocity varies linearly within the element:

$$\mathbf{v}_{E_i} = (v_j)_{j=1,\dots,d} = \begin{pmatrix} a_x + bx \\ a_y + by \end{pmatrix}. \quad (77)$$

Furthermore, we can compute the acceleration of the particle along its trajectory in the j-th direction by:

$$\left. \frac{dv_j}{dt} \right|_p = \frac{dv_j}{dx_j} \left. \frac{dx_j}{dt} \right|_p, \quad (78)$$

where t is time, and p denotes the position of the particle. The derivatives of the right-hand side of Equation 78 are:

$$\frac{dv_j}{dx_j} = b, \quad \left. \frac{dx_j}{dt} \right|_p = v_j. \quad (79)$$

Combining Equations 78, and 79, and given two distinct particle positions p_1 and p_2 on a trajectory within an element separated by a travel time Δt yields:

$$\ln\left(\frac{v_{j,p2}}{v_{j,p1}} \right) = \ln\left(\frac{a_j + b x_{j,p2}}{a_j + b x_{j,p1}} \right) = b \Delta t. \quad (80)$$

On every face, $F \in \mathcal{F}_{E_i}$, of an element E_i a vectorized line equation for every point, $\mathbf{x}_l = (x_l, y_l)$ on F, can be defined independent of the orientation of the line including the respective face:

$$\mathbf{x}_l = \mathbf{x}_{n1} + s\mathbf{t}_l, \tag{81}$$

where s is a scaling factor, and $\mathbf{t}_l = (t_{x,l}, t_{y,l})$ is a tangential unit normal vector parallel to the respective face, and starting from one facial node, $\mathbf{x}_{n1} = (x_{n1}, y_{n1})$, pointing towards the second facial node, $\mathbf{x}_{n2} = (x_{n2}, y_{n2})$. Furthermore, we know that for two points p_1 and p_2 on a trajectory, the travel time Δt in Equation 80 is the same no matter which spatial dimension x_j is considered, such that $\Delta t = \Delta t_x = \Delta t_y$, and the factor b in Equation 80 is element-wise constant. Considering these findings and Equation 80, we observe that:

$$\frac{a_x + bx_{e,p}}{a_x + bx_p} = \frac{a_y + by_{e,p}}{a_y + by_p}, \tag{82}$$

where $\mathbf{x}_{e,p} = (x_{e,p}, y_{e,p})$ is a possible exit point of a particle on a face, and $\mathbf{x}_p = (x_p, y_p)$ is an entry point of that particle. Then a possible exit-point on a face is given by substituting Equation 81 into Equation 82 yielding the scalar factor s such that Equation 81 points to the possible exit point $\mathbf{x}_{e,p}$ on a face:

$$s = \left(\frac{a_y + by_{n1}}{a_y + by_p} - \frac{a_x + bx_{n1}}{a_x + bx_p}\right) \cdot \left(\frac{bt_{x,l}}{a_x + bx_p} - \frac{bt_{y,l}}{a_y + by_p}\right)^{-1}. \tag{83}$$

Then, the vector of travel times $\boldsymbol{\tau} = (\Delta t)_{F \in \mathcal{F}_{E_i} \setminus \{F_s\}}$ for divergent flow on triangles is given by evaluating the travel times in an arbitrary spatial dimension:

$$\boldsymbol{\tau} = \left(\frac{1}{b} \ln\left(\frac{a_x + bx_{e,p}}{a_x + bx_p}\right)\right)_{F \in \mathcal{F}_{E_i} \setminus \{F_s\}} = \left(\frac{1}{b} \ln\left(\frac{a_y + by_{e,p}}{a_y + by_p}\right)\right)_{F \in \mathcal{F}_{E_i} \setminus \{F_s\}}. \tag{84}$$

Finally, the actual exit-point is determined according to Equation 76.

2.6.2 Analytical Solutions for Tetrahedra

Particle tracking for non-divergent flow on tetrahedra On tetrahedra the velocity is again element-wise constant for non-divergent flow and a possible exit point of a particle on a face is given by simple line-plane intersection. Again, let $\mathbf{x}_p = (x_p, y_p, z_p)$ be the starting point of a particle in an element E_i, or on a face $F_s \in \mathcal{F}_{E_i}$, and let $\boldsymbol{v}_{n,E_i} = (v_{x,n}, v_{y,n}, v_{z,n})$ be a vector of unit length pointing in the same direction as the element-wise velocity vector, $\mathbf{v}_{E_i}$, and let $\mathbf{n}_e = (n_{x,e}, n_{y,e}, n_{z,e})$ be a normal vector on a possible exit-face, $F \in \mathcal{F}_{E_i} \setminus \{F_s\}$. Furthermore, let $\mathbf{d} = \mathbf{p}_F - \mathbf{x}_p = (d_x, d_y, d_z)$ be a distance vector between the entry point $\mathbf{x}_p$ and an arbitrary point, $\mathbf{p}_F = \mathbf{x} \in F$, on the respective face. Given the following scalar factor:

$$d_l = \frac{\langle \mathbf{d}, \mathbf{n}_e \rangle}{\langle \mathbf{v}_{n,E_i}, \mathbf{n}_e \rangle}, \tag{85}$$

then, the possible exit-point on a face is given by:

$$\mathbf{x}_{e,p} = \mathbf{x}_p + d_l \mathbf{v}_{n,E_i}. \tag{86}$$

The vector of travel times is:

$$\boldsymbol{\tau} = \left(\frac{||\mathbf{x}_d||_2}{||\mathbf{v}_{E_i}||_2} \right)_{F \in \mathcal{F}_{E_i} \setminus \{F_s\}} \text{with} \operatorname{sign}(x_j) = \operatorname{sign}(v_{j,E_i}), \tag{87}$$

where $\mathbf{x}_d = \mathbf{x}_{e,p} - \mathbf{x}_p$. The actual exit point is again determined with Equation 76.

Particle Tracking for Divergent Flow on Tetrahedra For divergent flow a particle tracking scheme on tetrahedra can be derived in analogy to those for triangles. On tetrahedra the element-wise velocity field, with $\mathbf{x} = (x, y, z)$, is given by:

$$\mathbf{v}_{E_i} = (v_j)_{j=1,\dots,d} = \begin{pmatrix} a_x + bx \\ a_y + by \\ a_z + bz \end{pmatrix}, \tag{88}$$

and every face is embedded in a plane equation such that:

$$\mathbf{x}_l = \mathbf{x}_{n1} + s\mathbf{p}_1 + t\mathbf{p}_2, \tag{89}$$

where $\mathbf{x}_l = (x_l, y_l, z_l)$ is the vector of coordinates of an arbitrary point on a face F, $x_{n1} = (x_{n1}, y_{n1}, z_{n1})$ is the vector of coordinates of one of the nodes defining F, $\mathbf{p}_1 = (p_{x1}, p_{y1}, p_{z1})$ and $\mathbf{p}_2 = (p_{x2}, p_{y2}, p_{z2})$ are direction vectors of unit length parallel to F, which have to be linearly independent. For every point of a trajectory of a particle p, we again know that the travel time is the same no matter which spatial direction we consider $\Delta t = \Delta t_x = \Delta t_y = \Delta t_z$. Reconsidering Equation 80, we observe that:

$$\frac{a_x + bx_{e,p}}{a_x + bx_p} = \frac{a_y + by_{e,p}}{a_y + by_p} = \frac{a_z + bz_{e,p}}{a_z + bz_p}, \tag{90}$$

where $\boldsymbol{x}_{e,p} = (x_{e,p}, y_{e,p}, z_{e,p})$ is again a possible exit point of a particle on a face, and $\mathbf{x}_p = (x_p, y_p, z_p)$ is an entry point of a particle. Obviously, there are three possibilities to compute s for $d = 3$. We opt for s_1 such that $\Delta t_x = \Delta t_y$ which yields:

$$s_1 = \left(\frac{a_y + by_{n1} + btp_{y2}}{a_y + by_p} - \frac{a_x + bx_{n1} + btp_{x2}}{a_x + bx_p} \right) \cdot \underbrace{\left(\frac{bp_{x1}}{a_x + bx_p} - \frac{bp_{y1}}{a_y + by_p} \right)^{-1}}_{=:K}, \tag{91}$$

furthermore, s_2 is set such that $\Delta t_x = \Delta t_z$ which is then given by:

$$s_2 = \left(\frac{a_z + bz_{n1} + btp_{z2}}{a_z + bz_p} - \frac{a_x + bx_{n1} + btp_{x2}}{a_x + bx_p} \right) \cdot \underbrace{\left(\frac{bp_{x1}}{a_x + bx_p} - \frac{bp_{z1}}{a_z + bz_p} \right)^{-1}}_{=:L}. \tag{92}$$

The factor t is computed by $s_1 = s_2$ yielding:

$$t = \left(\frac{a_z + bz_{n1}}{(a_z + bz_p)\, L} - \frac{a_x + bx_{n1}}{(a_x + bx_p)\, L} - \left(\frac{a_y + by_{n1}}{(a_y + by_p)\, K} - \frac{a_x + bx_{n1}}{(a_x + bx_p)\, K} \right) \right) \cdot \left(\frac{bp_{y2}}{(a_y + by_p)\, K} - \frac{bp_{x2}}{(a_x + bx_p)\, K} - \left(\frac{bp_{z2}}{(a_z + bz_p)\, L} - \frac{bp_{x2}}{(a_x + bx_p)\, L} \right) \right)^{-1}. \tag{93}$$

Possible exit-points $\mathbf{x}_{e,p}$ on planes including the faces $F \in \mathcal{F}_{E_i} \setminus \{F_s\}$ for an element E_i, are then uniquely determined by inserting t (Equation 93) and either s_1 (Equation 91), or s_2 (Equation 92) for s in the plane equation (Equation 89) for a face F. It is obvious that s_1 (Equation 91), s_2 (Equation 92), and t (Equation 93) have, in any case, to be defined such that gaps in definition are avoided. That is, it has always to be checked, if the direction vectors, $\mathbf{t}_l$ for triangles, and $\mathbf{p}_1$, and $\mathbf{p}_2$ for tetrahedra are chosen such that divisions by zero are avoided.

The vector of travel times, $\boldsymbol{\tau} = (\Delta t)_{F \in \mathcal{F}_{E_i} \setminus \{F_s\}}$, for divergent flow on tetrahedra is again given by evaluating the travel times in an arbitrary spatial dimension:

$$\begin{aligned} \boldsymbol{\tau} &= \left(\frac{1}{b} \ln \left(\frac{a_x + bx_{e,p}}{a_x + bx_p} \right) \right)_{F \in \mathcal{F}_{E_i} \setminus \{F_s\}} = \left(\frac{1}{b} \ln \left(\frac{a_y + by_{e,p}}{a_y + by_p} \right) \right)_{F \in \mathcal{F}_{E_i} \setminus \{F_s\}} \\ &= \left(\frac{1}{b} \ln \left(\frac{a_z + bz_{e,p}}{a_z + bz_p} \right) \right)_{F \in \mathcal{F}_{E_i} \setminus \{F_s\}}. \end{aligned} \tag{94}$$

Finally, the actual exit-point is determined according to Equation 76.

2.6.3 The $\mathcal{RTN}_0$-Space in Mixed Coordinates on Triangular Prisms

A triangular prism can be represented as a superposition of two simplices: a line and a triangle. While element-wise analytical particle tracking on simplices is feasible in both global and local coordinates, we need to employ local coordinates when analyzing the velocity field in deformed non-simplicial elements. Upon transformation from global to local coordinates, all arbitrarily deformed triangular prisms become the same reference prism. In subsurface flow modeling, however, it is common practice to start the discretization with triangulation in the horizontal coordinates followed by expanding the triangles at the horizontal nodes in the vertical direction. This allows defining a reference prism in mixed coordinates, in which the horizontal local coordinates equal the corresponding global coordinates, while the vertical coordinates are mapped to the unit-interval, and therefore, scaled between 0 at the bottom of the prism and 1 at the top. Thus, our reference prism is not fixed, but it combines the triangular base area in global coordinates, mapped to local coordinates of the same value, with a unit height. By this, we have separated the horizontal from the vertical dimension. We can still conduct horizontal particle tracking in global coordinates on triangles, while tracking the particles in the vertical within the unit-interval. Then, the three-dimensional trajectory is given by direct superposition.

The $\mathcal{RTN}_0$ velocity space is a lowest-order velocity approximation within the individual elements yielding a distinct velocity vector in every spatial coordinate (Nédélec, 1980, 1986; Raviart

and Thomas, 1977). In $\mathcal{RTN}_0$-space, the spatial dimensions are independent of each other yielding, together with a constant normal component of the velocity on the element boundary for the reference element, a bilinear approximation of the velocity within the triangular prisms.

Before describing the actual velocity space, a few prerequisites on transformations of coordinates are necessary. In the following, we use $\mathbf{x}$ to denote the physical coordinates in three spatial dimensions $\mathbf{x} = (x, y, z)$. Furthermore, we use $\mathbf{X}$ to denote the coordinates in a local coordinate system associated to a reference element, such that the three spatial dimensions are $\mathbf{X} = (X, Y, Z)$. For the transformation of coordinates, we need the Jacobian matrix $\mathbf{J}$ of partial derivatives:

$$J_{i,j} = \frac{\partial x_i}{\partial X_j}, \tag{95}$$

$$\Rightarrow \mathbf{J} = \nabla_X \mathbf{N} \cdot \mathbf{x}_{pri}, \tag{96}$$

where i ranges over the dimensions in physical, global coordinates, and j ranges over the dimensions in local coordinates, $\nabla_X \mathbf{N}$ is the gradient of the shape functions $\mathbf{N}$ defining the reference prism in local coordinates, and $\mathbf{x}_{pri}$ contains the global coordinates of all nodes of the physical prism. The gradient of the shape functions in global coordinates, $\nabla_x \mathbf{N}$, can now be determined by:

$$\nabla_x \mathbf{N} = \mathbf{J}^{-1} \nabla_X \mathbf{N}. \tag{97}$$

For the integration over the element volume, the following rule applies:

$$\int_{E_x} d\mathbf{x} = \int_{E_X} \det(\mathbf{J}) d\mathbf{X}, \tag{98}$$

where E_x is an element in global, and E_X is an element in local coordinates, respectively. The same rule applies to integrations within the element, like the pathline-integration employed in particle tracking. Furthermore, the transformation of coordinates is linear such that (e.g., Rognes et al., 2009):

$$\mathbf{x} = \mathbf{J}\mathbf{X} + \mathbf{c}. \tag{99}$$

Considering the transformation shown in Equation 99, we also see that the barycentric weights on a simplex in local coordinates are the same for every point in the element as in global coordinates, which is a useful attribute one may exploit, for example, for mapping particle locations from physical to local coordinates. The described transformation from local to global coordinates is often called Piola transformation; it is a standard procedure in finite element simulations of scalar state variables like hydraulic heads, approximated at the nodes and interpolated within the element. For mapping a vector-field onto a space, which either preserves the divergence or the curl, special rules for transforming normal and tangential traces on the element faces apply. These rules are explained in great detail by Rognes et al. (2009), on whose descriptions

and derivations we rely. As particle tracking is based on conforming velocity fields implying mass-conservation, we opted for the lowest-order velocity space which preserves the divergence, facilitating the derivation of analytical solutions of trajectories in the elements. Contravariant Piola transformation applies to the normal traces of the velocity in $\mathcal{RTN}_0$-space. Considering this, Rognes et al. (2009) showed that the following transformation rule is valid:

$$\underbrace{|F|\boldsymbol{\phi}(\mathbf{x})\cdot\mathbf{n}}_{\text{in global coordinates}} = \underbrace{|F_0|\boldsymbol{\Phi}(\mathbf{X})\cdot\mathbf{n_0}}_{\text{in local coordinates}}, \tag{100}$$

where $|F|$ is the measure of a face in global coordinates, $|F_0|$ is the measure of the same face in local coordinates, $\mathbf{n}$ is the normal vector on F in global coordinates pointing outwards, and $\mathbf{n_0}$ is the same normal vector in local coordinates. $\boldsymbol{\phi}(\mathbf{x})$ is a vector-valued function approximated in some velocity space on the elements in global coordinates, and $\boldsymbol{\Phi}(\mathbf{X})$ is the equivalent vector-valued function in local coordinates. From Equation 100 one directly sees that transforming the normal component of a vector field on an element boundary from one coordinate system to the other reduces to the multiplication with a scaling factor equal to the ratio of the measures of the faces in the two coordinate systems.

As mentioned above, we consider prisms with a constant triangular base area in the horizontal directions and linearly varying height. We opt for a reference prism in mixed coordinates, in which the horizontal coordinates have the same value as in global horizontal coordinates without the unit, whereas the vertical extension of the prism is scaled to the dimensionless unit interval $Z = [0, 1]$. For this choice of local coordinates, the determinant of the Jacobian becomes:

$$\det(\mathbf{J}(X, Y)) = h_E(x, y), \tag{101}$$

where h_E is the scalar value of the height of the prism at the horizontal coordinates (x, y) although having dimensions of $[\mathrm{L}^3]$. This gives an intuitive measure on the magnitude of variation of $\det(\mathbf{J})$ within the element. Also, particle tracking in the horizontal directions can be conducted in global coordinates saving transformations of coordinates, mapping of faces, and associated bookkeeping.

For a triangular prism E_i the specific-discharge vector $\mathbf{q}_{E_i,mixed} = (q_{x,i}, q_{y,i}, q_{Z,i})$ in mixed coordinates can be computed by:

$$q_{x,i} = \sum_{F_{tri}\subset\partial E_{tri}} \frac{|F_{tri}|}{2|E_{tri}|}\,(x - P_{x,F_{tri}})\cdot q^{\perp,i}_{h,F_{tri}} = a_{x,i} + b_{h,i}x, \tag{102}$$

$$q_{y,i} = \sum_{F_{tri}\subset\partial E_{tri}} \frac{|F_{tri}|}{2|E_{tri}|}\,(y - P_{y,F_{tri}})\cdot q^{\perp,i}_{h,F_{tri}} = a_{y,i} + b_{h,i}y, \tag{103}$$

$$q_{Z,i} = \frac{(Z-1)\cdot q^{\perp,i}_{bottom}\cdot\frac{|F^{bottom}|}{|F^{bottom}_0|} + Z\cdot q^{\perp,i}_{top}\cdot\frac{|F^{top}|}{|F^{top}_0|}}{h_{E_i}(x,y)} = \frac{a_{Z,i} + b_{Z,i}Z}{h_{E_i}(x,y)}, \tag{104}$$

where $|F_{tri}|$ is the length of an edge F_{tri} of the base triangle of E_i, $|E_{tri}|$ is the area of the

base triangle, and $P_{F_{tri}} = (P_{x,F_{tri}}, P_{y,F_{tri}})$ are the horizontal coordinates of the node opposite to the edge F_{tri} in the horizontal plane of the element E_i; $|F^{bottom}|$ and $|F^{top}|$ are the areas of the bottom face F^{bottom} and the top face F^{top} of the physical element, whereas $|F_0^{bottom}|$ and $|F_0^{top}|$ are the areas of the bottom face F_0^{bottom} and the top face F_0^{top} in local coordinates, both $|F_0^{bottom}|$ and $|F_0^{top}|$ are equal to area of the base triangle, $|E_{tri}|$, in mixed coordinates; $q_{h,F_{tri}}^{\perp,i}$, $q_{bottom}^{\perp,i}$, and $q_{top}^{\perp,i}$ are the specific discharge-values in global coordinates computed by the finite-volume flux reconstruction normal to the side edge F_{tri}, the bottom and top face of the element i, respectively. All normal flux components $q^{\perp,i}$ are also equipped with a sign convention such that: $q_F^{\perp,i} = (\boldsymbol{\nu}_F \cdot \mathbf{q}) \cdot (\mathbf{n}_{E_i,F} \cdot \boldsymbol{\nu}_F)$, in which F can be the edge F_{tri}, the bottom, or the top face of the element E_i. Such a sign convention is common among adjacent elements. The coefficients $a_{x,i}$, $a_{y,i}$, $b_{h,i}$ describe the linear variation of the horizontal specific-discharge component in an element i and can be computed by summation. Note that the horizontal specific discharge components $q_{x,i}$ and $q_{y,i}$ in the Equations 102 and 103 are given in global, physical coordinates having dimensions of $[\mathrm{LT}^{-1}]$ while the vertical specific discharge component q_Z in Equation 104 is given in the local coordinates of the mixed reference prism having dimensions of an inverse time. If the Equations 102 and 103 were formulated in the local coordinates of the mixed reference prism the horizontal specific discharge components also would have dimensions of an inverse time while the values of all variables would remain the same although being dimensionless in space.

While there is a linear dependence of q_Z with Z, the linear variation of the height and, therefore, of the determinant of the Jacobian, $h_E(x, y)$, in the horizontal coordinates leads to a complicated dependence of q_Z on the mixed coordinates (x, y, Z).

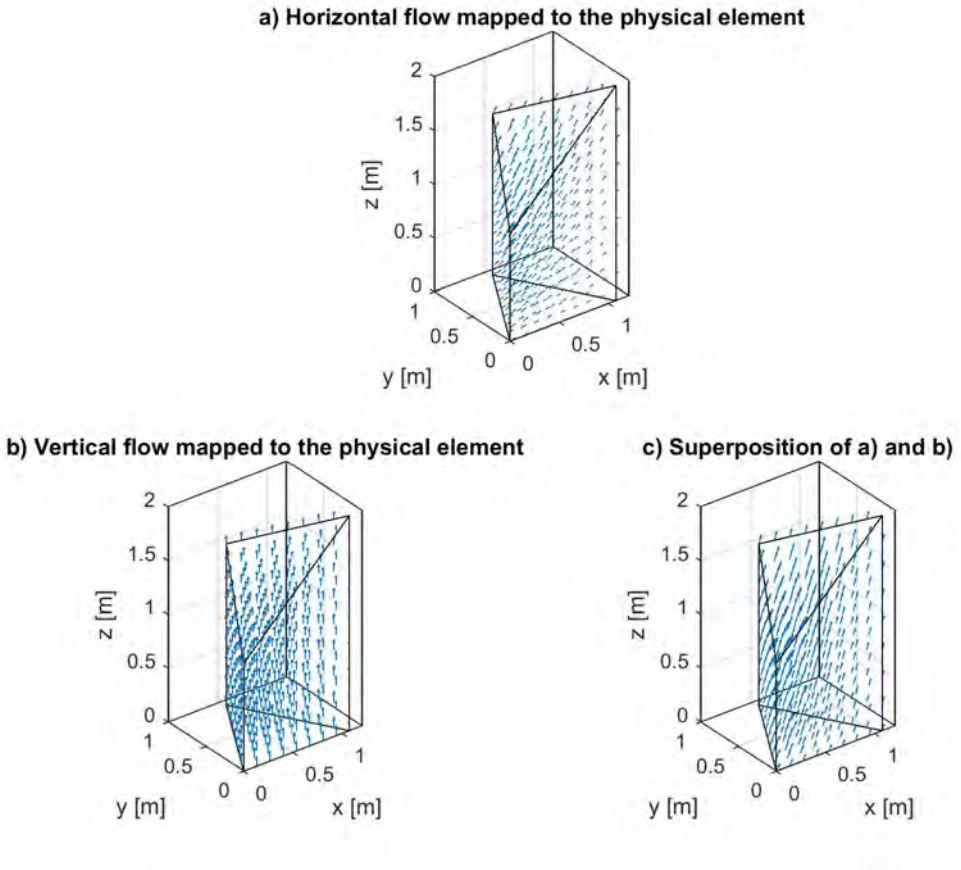

Figure 5: Exemplification of a velocity field in $\mathcal{RTN}_0$-space. a) Non-divergent horizontal velocity field mapped to the physical element. Note that, although there is a vertical component of the physical velocity above the bottom face, the field is topologically horizontal and the vertical component is 0 in mixed coordinates. The normal component of the velocity on the top and bottom faces is zero in mixed coordinates as well as in the physical element; b) divergent vertical flow field; c) superposition of a) and b). (This Figure is taken from Selzer et al. (2021).)

The specific-discharge vector $\mathbf{q}_{E_i,mixed}$, now given in local coordinates in every dimension, can be transformed into the specific-discharge vector in global coordinates $\mathbf{q}_{E_i}$ by:

$$\mathbf{q}_{E_i} = \mathbf{J} \cdot \mathbf{q}_{E_i,mixed}. \tag{105}$$

2.6.4 Analytical Approximation for Triangular Prisms

Recall, that for a given velocity field $\mathbf{v}(\mathbf{x}, t)$, particle trajectories $\mathbf{x}_p(t)$ and associated travel times τ are computed by:

$$\mathbf{x_p}(t) = \mathbf{x}_{p,t_0} + \int_{t_0}^{t} \mathbf{v}(\tau) d\tau, \tag{106}$$

$$\tau = \int_S \frac{1}{\|\mathbf{v}(\mathbf{s})\|_2} d\mathbf{s}, \tag{107}$$

where $\mathbf{x}_{p,t_0}$ is the vector of starting coordinates, S is the (curvilinear) trajectory, while s denotes

all coordinates defining S, and $\mathbf{v}$ denotes the linear average velocity, which determines the particle trajectory S, and which is related to the specific-discharge vector $\mathbf{q}$ by:

$$\mathbf{v} = \frac{\mathbf{q}}{\theta}. \tag{108}$$

In the following, we assume that the volumetric water content θ is constant within an element. In our scheme, this constant water content is the arithmetic mean of the volumetric water contents at the nodes of the element of interest, which equals a linear interpolation to the centroid.

Given a conforming velocity field in an appropriate velocity space, particle tracking can either be computed by numerical integration (e.g., Matringe et al., 2006; Srinivasan and Lipnikov, 2013) or analytically (e.g., Crane and Blunt, 1999; Loschko et al., 2016; Matringe et al., 2006; Pollock, 1988; Prévost et al., 2002; Selzer and Cirpka, 2020). The key idea of the semi-analytical methods is to evaluate the particle trajectory analytically based on an element-wise given velocity field, which is numerically approximated. In this approach, the exit point and travel time are determined from the entry point by closed-form integration of the kinematic equation using the velocity approximation within the element. The particle is then tracked further in the neighboring element until it has reached an outflow boundary or the total travel time to be simulated has been reached. The advantages of the semi-analytical approach is that the integration within the elements is exact.

Equations 102 and 103 indicate that, in our flux approximation on deformed prisms, the horizontal specific-discharge components q_x and q_y within an element vary linearly in their respective direction, while they are constant in all other directions. Because we assume that the volumetric water content is constant within the element, the same properties hold for the horizontal components of the linear average velocity v_x and v_y.

Equation 104 shows that the local vertical specific discharge q_Z on deformed prisms depends not only on the local vertical coordinate Z, but also on the horizontal coordinates, which complicates the analytical evaluation of vertical displacements. Rather than performing a numerical integration, we decided to simplify Equation 104 by the following approximation (for an analogous simplification compare Prévost et al., 2002):

$$q_{Z,i} \approx \frac{(Z-1)\cdot q^{\perp,i}_{bottom}\cdot\frac{|F^{bottom}|}{|F^{bottom}_0|} + Z\cdot q^{\perp,i}_{top}\cdot\frac{|F^{top}|}{|F^{top}_0|}}{h_{E_i}(x_{c,i},y_{c,i})} = a_{*Z,i} + b_{*Z,i}Z, \tag{109}$$

where $x_{c,i}$ and $y_{c,i}$ are the horizontal coordinates of the element centroid of the element E_i, $\mathbf{x}_{c,i} = (x_{c,i}, y_{c,i})$. With this, the three spatial specific-discharge components are decoupled.

2.6.5 Algorithmic Details

In our particle tracking code, we track particles, originating form an arbitrary starting location within the domain, from one element boundary to the next. Given a starting location of a particle, we first search for the starting element. If particles are tracked, e.g., only from Dirichlet

boundary conditions, we can also restrict the search to the respective elements sharing a Dirichlet boundary, or the respective starting elements are known beforehand. This may be the case, if particles are released at the top of the domain. If the start location is at an element interface or a distinct node, we pragmatically choose the first possible starting element found in the search. Details on the search employing barycentric coordinates are given at the end of this section. If the starting element is determined, all the following elements are determined, by tracking a particle from face to face. This scheme is in analogy to that of Pollock (1988). According to some criterion reflecting numerical accuracy, our particle tracking algorithm automatically switches between a tracking for non-divergent and divergent flow, depending on the properties of the element-wise flow field. The tracking procedure is continued until a particle leaves the domain via a Dirichlet boundary, an outflow boundary, or an element containing an extraction well, or the desired simulation time is reached.

The semi-analytical approach of Pollock (1988) relies on decoupled velocity components within elements. In our adaptation, it consists of the following steps exemplified for triangular prisms:

1. Extend the faces of the element to infinite planes.

2. For a starting location $\mathbf{x}_0$ within the element, or at an inflow face of the element, determine which faces are potential exit faces for the particle. The velocity component normal to such a face must be oriented outwards of the element.

3. For a given element i and each potential exit plane m, determine the travel time $\tau_{i,m}$ it would take to reach that plane by analytical integration for a dimension $d \in \{x, y, *Z\}$:

$$\tau_{i,m} = \begin{cases} \dfrac{\theta_i (x_{i,m,d} - x_{0,d})}{a_{d,i}}, & \text{if } |b_{d,i}| < \varepsilon_b, \\ \dfrac{\theta_i}{b_{d,i}} \ln\left(\dfrac{a_{d,i} + b_{d,i} x_{i,m,d}}{a_{d,i} + b_{d,i} x_{0,d}} \right), & \text{otherwise,} \end{cases} \tag{110}$$

   where $a_{d,i} \in \{a_{x,i}, a_{y,i}, a_{*Z,i}\}$ and $b_{d,i} \in \{b_{h,i}, b_{*Z,i}\}$ are the coefficients describing the linear dependence of the specific-discharge component in its dimension d in the element E_i, $x_{i,m,d}$ is the possible exit location on the face F_m in dimension d. Note that, if E_i takes a sink not projected on ∂E_i, like it would be the case for an extraction well, the exit location $x_{i,m,d}$ could be also within the element. $x_{0,d} \in \mathbf{x}_0 = \{x_0, y_0, *Z_0\}$ is the coordinate of the starting location $\mathbf{x}_0$ being in element E_i or on ∂E_i, and θ_i is the volumetric water content associated to element i obtained by linear interpolation of the nodal water contents to the centroid of element i. Note that $\tau_{i,m}$ is the same for a possible exit location for every dimension d. ε_b is a small number on the order of 100 times the relative machine precision.

4. The actual travel time τ_i within the element is now the minimum positive and real travel time towards all face-carrying planes or the center of an internal sink.

5. With the given travel time τ_i, the exit point $\mathbf{x}_{ex} = (x_{ex}, y_{ex}, *Z_{ex})$ in mixed coordinates is

now given by:

$$x_{ex} = \begin{cases} x_0 + \dfrac{a_{x,i}}{\theta_i}\tau_i, & \text{if } |b_{h,i}| < \varepsilon_b, \\ \left(x_0 + \dfrac{a_{x,i}}{b_{h,i}}\right)\exp\left(\dfrac{b_{h,i}}{\theta_i}\tau_i\right) - \dfrac{a_{x,i}}{b_{h,i}}, & \text{otherwise,} \end{cases} \tag{111}$$

$$y_{ex} = \begin{cases} y_0 + \dfrac{a_{y_i}}{\theta_i}\tau_i, & \text{if } |b_{h,i}| < \varepsilon_b, \\ \left(y_0 + \dfrac{a_{y,i}}{b_{h,i}}\right)\exp\left(\dfrac{b_{h,i}}{\theta_i}\tau_i\right) - \dfrac{a_{y,i}}{b_{h,i}}, & \text{otherwise,} \end{cases} \tag{112}$$

$$*Z_{ex} \approx \begin{cases} *Z_0 + \dfrac{a_{*Z,i}}{\theta_i}\tau_i, & \text{if } |b_{*Z,i}| < \varepsilon_b, \\ \left(*Z_0 + \dfrac{a_{*Z,i}}{b_{*Z,i}}\right)\exp\left(\dfrac{b_{*Z,i}}{\theta_i}\tau_i\right) - \dfrac{a_{*Z,i}}{b_{*Z,i}}, & \text{otherwise.} \end{cases} \tag{113}$$

6. Back-transform the local vertical coordinate $*Z_{ex}$ to the corresponding physical coordinate z_{ex}.

7. Determine whether the particle has reached an outflow boundary of the domain, or the center of an internal sink. If not, analyze which neighboring element E_j will be entered, switch to that element such that E_j becomes E_i, set $\mathbf{x}_0 = \mathbf{x}_{ex}$, and repeat all steps.

Selzer and Cirpka (2020) give analytical expressions on how to perform these calculations most efficiently for triangles (compare section 2.6.1). These expressions directly apply for the horizontal tracking in the triangular prisms.

Depending on the application, we keep track of the full particle trajectory, keep only travel times and the type of boundary where the particle exits the domain, or travel times spent in elements of specific material properties. By reverting the velocity field and the time arrow, we can also evaluate where particles have come from, how long they have been in the domain, and how much time they spent in certain zones.

Note that the described algorithm is fastest (at least when written in Matlab), if all computations are done within one big while-loop, which runs, as long as there is at least one particle, which has not finished its trajectory. Within the while-loop, there should be not further loop, for the particle tracking code on triangular prisms, the code was written by the author as such. All computations necessary for transient catchment-scale particle tracking are done loop-free employing among others vector-matrix multiplications and multidimensional indexing, which is all combined with propositions of mathematical logic, such that all special cases (including extraction wells, and even possible failure of a trajectory among others) are covered, and every particle moves with its correct velocity depending on its location in space and time. If there is a time-update because of transient flow, the particle stops exactly at this time, and is then further tracked with the new velocity.

Using Barycentric Coordinates to Find the Starting Element of a Particle Trajectory

The outlined scheme below is designed to determine, if a point is in a simplex. If an element is a polyhedron, like a triangular prism, this element can be divided into simplices of the same dimension. Then, the equations below can still be used to determine whether the starting location of a particle is within the respective element.

Every point $\hat{\mathbf{x}}_s$ within a simplex can be described by a linear combination of the coordinates of the nodes, $\hat{\mathbf{x}}_i$, and a barycentric, nodal weight, $\beta_i \geq 0$, respectively, such that:

$$\hat{\mathbf{x}}_s = \sum_{i \in \mathcal{N}_E} \beta_i \hat{\mathbf{x}}_i, \tag{114}$$

with

$$\sum_{i \in \mathcal{N}_E} \beta_i = 1, \quad \text{and} \quad \beta_j = 1 - \sum_{i \neq j \in \mathcal{N}_E} \beta_i, \tag{115}$$

where j is the index of a node and β_j is its associated weight, such that any point in the simplex can be described by the coordinates of its nodes, and $d+1$ weights. Combining Equations 114 and 115 leads to:

$$\mathbf{T}\boldsymbol{\beta} = \hat{\mathbf{x}}_s - \hat{\mathbf{x}}_j, \tag{116}$$

$$\Rightarrow \boldsymbol{\beta} = \mathbf{T}^{-1} \left(\hat{\mathbf{x}}_s - \hat{\mathbf{x}}_j \right), \tag{117}$$

where $\mathbf{T}$ is a transformation matrix only depending on the coordinates of the nodes, $\mathbf{T}^{-1}$ can easily be evaluated analytically. $\boldsymbol{\beta}$ is the vector of the d independent, barycentric, nodal weights, and $\hat{\mathbf{x}}_j$ is the vector of coordinates of node j of the element.

We exploit the concept of barycentric coordinates in the search for the element in which the starting point of our particle-tracking scheme resides. To do so, we define the particle location as $\mathbf{x}_p = \hat{\mathbf{x}}_s$ and solve Equation 117 for every element. If a particle lies within an element, all weights $\boldsymbol{\beta} = (\beta_i)_{i=1,\dots,d+1}$ are within the interval $0 \leq \beta_i \leq 1$ and sum up to unity. We stop the search at the first instance at which both criteria are met.

For finding starting points in triangular prisms, we assume that the prisms are only deformed in the vertical, while the horizontal base area stays constant for triangular prisms stacked on each other. In this case, we first search for the starting locations in the horizontal triangular base areas, followed by a search in the vertical dimension to identify the actual prism, in which the starting point of a particle is eventually located.

2.7 Measures of Differences between the Discretization Methods

We define difference measures comparing the specific discharge values at all element centroids of the $\mathcal{RTN}_0$-projection with a reference solution being either analytical or obtained by the

cell-centered FVM. The first difference measure ϵ_{abs} is the ratio between the absolute values:

$$\epsilon_{abs} = \frac{||\mathbf{q}_{E_i}^{RTN_0}(\mathbf{x}_c)||_2}{||\mathbf{q}_{E_i}^{Ref}(\mathbf{x}_c)||_2}, \quad E_i \in \mathcal{T}, \tag{118}$$

where $\mathbf{q}_{E_i}^{RTN_0}(\mathbf{x}_c)$ is the specific discharge evaluated at the centroid of element E_i, obtained by the $\mathcal{RTN}_0$-projection of The $\mathcal{P}_1$ Galerkin FEM solution, and $\mathbf{q}_{E_i}^{Ref}(\mathbf{x}_c)$ is the same specific discharge computed by the reference solution. The second difference measure quantifies the discrepancy in direction by evaluating the scaled angle between the specific discharge vectors:

$$\epsilon_{dir} = \frac{\angle\left(\mathbf{q}_{E_i}^{RTN_0}(\mathbf{x}_c), \mathbf{q}_{E_i}^{Ref}(\mathbf{x}_c)\right)}{\pi}, \quad E_i \in \mathcal{T}, \tag{119}$$

where $\angle(\cdot,\cdot)$ denotes the angle in radians between two vectors. For evaluating the overall behavior of a model, the element-wise measures ϵ_{abs} and ϵ_{dir} are assembled to the vectors $\boldsymbol{\epsilon}_{abs}$ and $\boldsymbol{\epsilon}_{dir}$ listing all element-wise values, respectively.

2.8 Implementation Details

The grids for the two-dimensional test cases are generated by the algorithm 'triangle' (Shewchuk, 2005), accessed via MeshPy using Python 3.4.2 employing the compiler GCC 4.9.1 on Linux Debian 8. The tetrahedral grids consist of manually defined cubes, which are subsequently split into triangular prisms and finally into tetrahedra. This splitting is done manually by a scheme described by the author of this thesis (see section 3.1.4). The splitting yields conforming grids, which also minimize grid effects in lowest-order finite-volume type discretizations. The primal hydraulic-head solutions are, as already mentioned, obtained by using HydroGeoSphere (Aquanty, Inc., 2015) on Windows 7 for the examples employing standard Galerkin FEM. However, any $\mathcal{P}_1$ Galerkin FEM code, and moreover any FEM code yielding a non-conforming velocity field, could be used, in principle. The primal solutions for the finite difference method in finite-element terms is obtained using HydroGeoSphere on Windows 10 (Aquanty, Inc., 2015; Therrien and Sudicky, 1996), and on Linux Ubuntu for the sub-catchment-scale example (see section 3.2.2), using the finite-difference mode of HydroGeoSphere, which employs the finite difference method in finite-element terms described in section 2.4.2. The cell-centered FVM, the $\mathcal{RTN}_0$-projection, the finite-volume flux reconstruction, and the particle tracking codes are implemented in Matlab 2016b, this framework reads mostly binary, and seldom ascii output/input files generated by HydroGeoSphere. For transient flow, the nodal volumetric water content can be reconstructed for every element using the van Genuchten parameterization (see Equation 9), or directly read in as binary file containing the water saturations of the primal solution on the nodes for every element. The final framework proposed for catchment-scale models of variably saturated flow is split into three major parts. The first computes all geometrical properties and has to be run once per grid. The second is the FVM flux reconstruction that is run once per time level for which a flux-reconstruction is desired. We solve the final linear system of

equations (Equation 68) by the stabilized biconjugate gradients method in Matlab using an incomplete LU factorization as preconditioner, and a starting vector computed from the primal hydraulic-head solution linearly interpolated to the centroids of the elements. The third part is the actual particle tracking routine, here the algorithm switches for every element individually between non-divergent and divergent flow according to a user-defined criterion reflecting numerical accuracy.

3 Numerical Results

3.1 Results for the $\mathcal{RTN}_0$-Projection

In the following sections numerical test cases for the $\mathcal{RTN}_0$-projection and associated particle trajectories are evaluated. With the exception of section 3.1.5 on an application of the $\mathcal{RTN}_0$-projection to variably saturated flow, and the descriptions on optimal splitting of structured grids of cubes into conforming grids of tetrahedra in section 3.1.4, the following content and text has been published in Selzer and Cirpka (2020), sometimes in slightly adapted form.

3.1.1 Non-Divergent Groundwater Flow on Triangles

We first compare the $\mathcal{RTN}_0$-projection to the cell-centered FVM, and the original $\mathcal{P}_1$ Galerkin velocity approximation for non-divergent groundwater flow in a square domain of dimension $100\,\mathrm{m} \times 100\,\mathrm{m}$ containing two nearly impervious walls. The discretized domain consists of 2554 elements and 1333 nodes (see Figures 6a-c). The left boundary of the domain is a Dirichlet boundary with a fixed head value of $\hat{h}_{D,l} = 11.0\,\mathrm{m}$. The right boundary of the domain is split. Over a central section ($x = 100$, $y \in [30, 70]$) the hydraulic head is fixed to $\hat{h}_{D,r} = 10.0\,\mathrm{m}$. All other boundary sections (top, bottom, remaining parts of the right-hand side boundary) are no-flow boundaries. Yellow shaded elements in Figures 6a-c have an isotropic hydraulic-conductivity value of $k = 10^{-4}\,\mathrm{m/s}$. The blue-gray shaded elements belong to the nearly impermeable walls in $x \in [30, 40]$, $y \in [0, 70]$ and $x \in [60, 70]$, $y \in [30, 100]$. They exhibit a reduced hydraulic-conductivity value of $k = 10^{-10}\,\mathrm{m/s}$. The uniform porosity is $n_e = 0.4$. Particles are started on the left-hand Dirichlet boundary with a spacing reflecting the cumulative flux across the boundary. For the case of non-divergent groundwater flow, the latter implies that the calculated particle trajectories are streamlines such that any two neighboring trajectories are streamtubes with the same discharge.

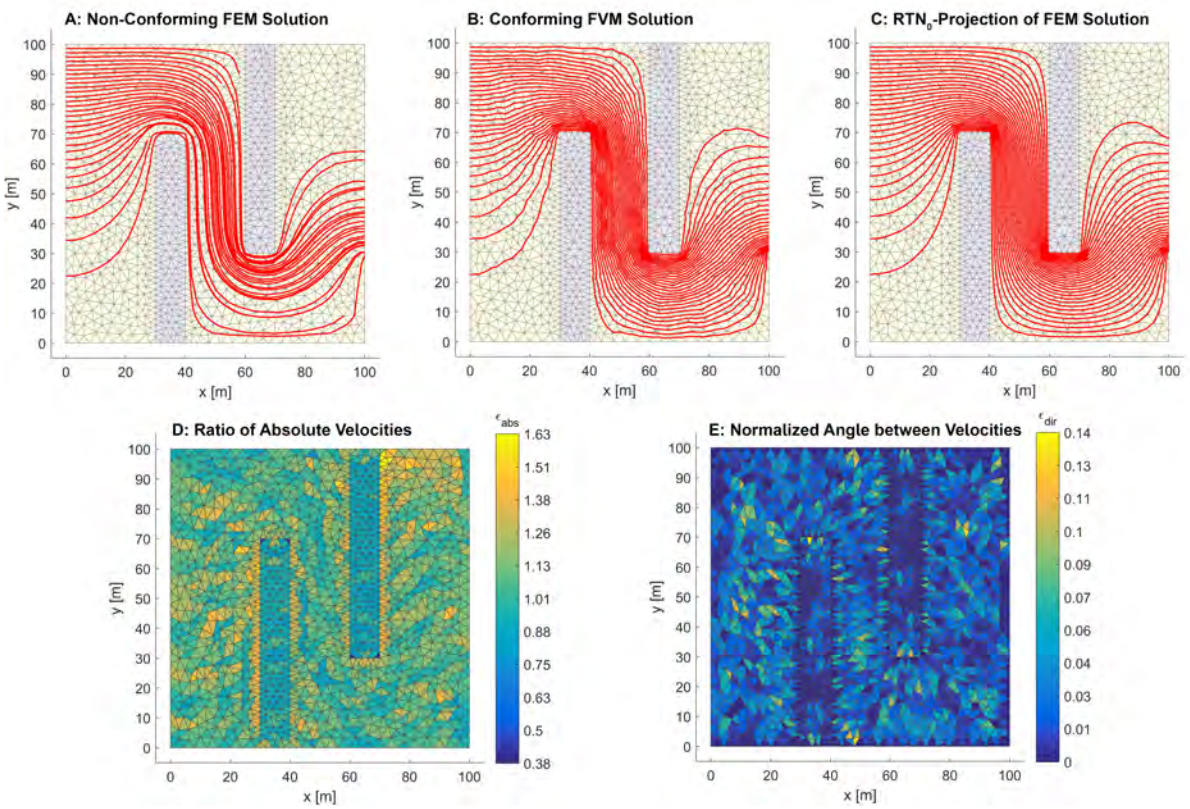

Figure 6: Particle trajectories and difference measures of the two conforming velocity fields for non-divergent groundwater flow in a 2-D domain with two nearly impermeable walls. a) trajectories based on non-conforming $\mathcal{P}_1$ Galerkin FEM velocity approximation; b) trajectories based on cell-centered FVM; c) trajectories based on the projection of the $\mathcal{P}_1$ Galerkin FEM velocity approximation onto a mass-conservative field in $\mathcal{RTN}_0$-space; d) ratio of absolute velocities at cell centroids ($\boldsymbol{\epsilon}_{abs}$) resulting from the proposed $\mathcal{RTN}_0$-projection of the non-conforming $\mathcal{P}_1$ Galerkin FEM velocity field and the cell-centered finite volume method; e) angle between the two velocities at cell centroids ($\boldsymbol{\epsilon}_{dir}$). (Adapted from Selzer and Cirpka (2020).)

Figure 6a shows particle trajectories using the element-wise velocity approximation $\mathbf{q}_E^{FE}$ based on the $\mathcal{P}_1$ Galerkin FEM. As discussed, specific-discharge and average linear velocity fields, $\mathbf{q}(\mathbf{x})$ and $\mathbf{v}(\mathbf{x})$, respectively, are non-conforming, causing jumps of the normal components on faces. The erroneous physical interpretation of these jumps are virtual sources and sinks on the faces. These numerical sources and sinks are preserved in the tracking patterns. In the most extreme case, the element-wise velocity vectors of neighboring elements both point towards the same face. In such cases, we could not track the particles any further and thus made the trajectories end on the respective face.

Figure 6b shows the particle trajectories of the conforming cell-centered FVM solution of the groundwater flow equation (Equation 3) on triangles, whereas Figure 6c shows the trajectories resulting from the $\mathcal{RTN}_0$-projection of the $\mathcal{P}_1$ Galerkin FEM velocity approximation. The trajectories of the cell-centered FVM solution are somewhat more angular with small rapid changes of the direction at cell interfaces. This is due to the strong constraints of the FVM discretization and has also been observed for mixed finite element solutions in $\mathcal{RTN}_0$-space (Putti and Sartoretto, 2009; Matringe et al., 2006; Juanes and Matringe, 2009; Hoteit et al.,

2002). In contrast, our $\mathcal{RTN}_0$-projection leads to smoother trajectories, which we assess as being more correct in sub-domains of uniform hydraulic conductivity. We believe that the smoothness of the $\mathcal{RTN}_0$-projection stems from the smoothness of the hydraulic gradients in the $\mathcal{P}_1$ Galerkin FEM solution. While the latter needs to be corrected in order to obtain a mass-conservative $\mathcal{RTN}_0$-solution of the velocity field, the correction is minimized within the constraints of a conforming, mass-conservative velocity approximation. Of course, the velocity fields of both the FVM-solution and our projection are in $\mathcal{RTN}_0$-space, are conforming and element-wise mass conservative. While the $\mathcal{RTN}_0$-projection of the $\mathcal{P}_1$ Galerkin FEM velocity approximation is smoother, the computational effort to obtain the cell-centered FVM solution is considerably lower.

Table 1: Measures for the differences between the discretization methods in the 2-D application with two nearly impervious walls. As $\boldsymbol{\epsilon}_{abs}$ is a ratio, we give the respective infinity norm as $||(|1-\boldsymbol{\epsilon}_{abs}|)||_\infty$. (Taken from Selzer and Cirpka (2020).)

	$mean(\cdot)$	$median(\cdot)$	$\|\|(\|\cdot\|)\|\|_\infty$
$\boldsymbol{\epsilon}_{abs}$	1.08	1.06	0.63
$\boldsymbol{\epsilon}_{dir}$	$2.30 \cdot 10^{-2}$	$1.70 \cdot 10^{-2}$	$1.43 \cdot 10^{-1}$

We quantify the difference between the two conforming velocity fields depicted in Figures 6b and 6c by the ratio of absolute velocities according to Equation 118 and the angle between the two velocities according to Equation 119. Figures 6d and 6e show the spatial distribution of these two difference measures, and Table 1 lists norms over the entire domain. From this, we conclude that the velocities originating from our $\mathcal{RTN}_0$-projection are very similar to those directly obtained from cell-centered FVM, with respect to both the absolute values and directions of the velocities throughout the domain, including the zones within or nearby the nearly impervious walls of very low hydraulic conductivity. In particular, the $\mathcal{RTN}_0$-projection does not introduce a noticeable numerical rotation, which might occur in other flux-correction schemes (Schiavazzi, 2013), and occasionally also within the normal framework of mixed finite elements in $\mathcal{RTN}_0$-space (Hoteit et al., 2002). Note that even though both solutions are in $\mathcal{RTN}_0$-space, one would expect different results as they are computed differently. As both solutions are numerical approximations, we cannot say which solution is more accurate. In general, the difference measures depicted in Table 1 are all very low, and also depend on numerical accuracy of the computations and the solver used.

3.1.2 Divergent Groundwater Flow on Triangles

In our next benchmark, we consider a $100\,\mathrm{m} \times 100\,\mathrm{m}$ domain with uniform isotropic hydraulic conductivity of $k = 10^{-5}\,\mathrm{m/s}$, in which groundwater recharge takes place in a $30\,\mathrm{m} \times 30\,\mathrm{m}$ squared subdomain in the center of the domain, indicated by a blue-gray shaded square in Figure 7. The recharge flux is $q_{in} = 200\,\mathrm{mm/a} \approx 6.3376 \cdot 10^{-9}\,\mathrm{m/s}$. The uniform porosity is $n_e = 0.4$. The domain is discretized by 1252 triangular elements and 667 nodes. The left and right boundaries of the domain are Dirichlet boundaries with a head difference of 1.0 m, whereas

no flow is allowed across the upper and lower boundaries.

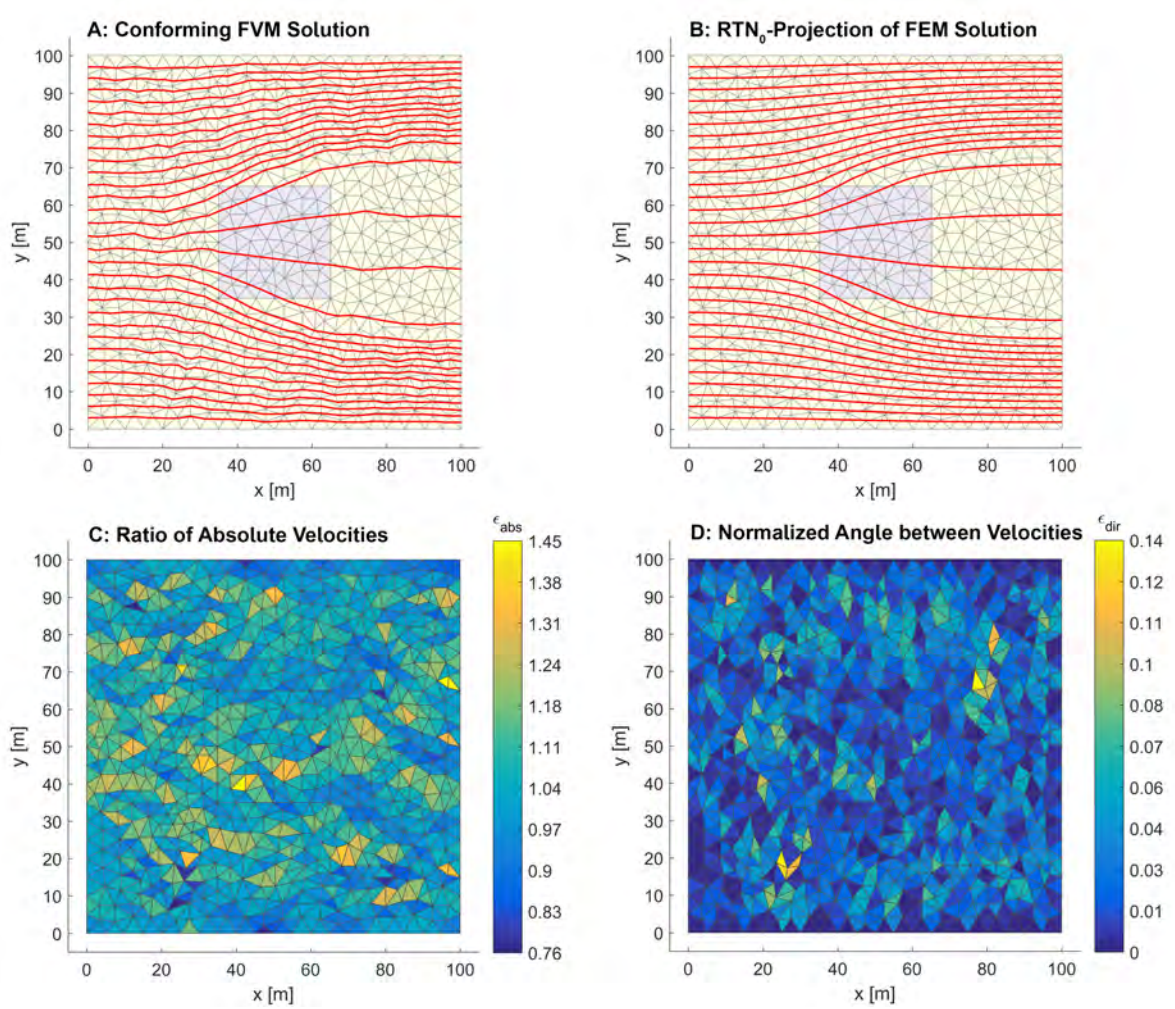

Figure 7: Particle trajectories and difference measures comparing the velocity fields resulting from the $\mathcal{RTN}_0$-projection of a non-conforming field with the velocity field computed by cell-centered FVM for a 2-D domain with a recharge zone. a) trajectories for cell-centered FVM; b) trajectories for the $\mathcal{P}_1$ Galerkin FEM velocity approximation projected onto a field in $\mathcal{RTN}_0$-space; c) ratio of absolute velocities at cell centroids ($\boldsymbol{\epsilon}_{abs}$); d) angle between the velocities at cell centroids ($\boldsymbol{\epsilon}_{dir}$). (Adapted from Selzer and Cirpka (2020).)

Table 2: Measures for the differences between the discretization methods in the 2-D application with a rectangular recharge zone. As $\boldsymbol{\epsilon}_{abs}$ is a ratio, we give the respective infinity norm as $||(|1 - \boldsymbol{\epsilon}_{abs}|)||_\infty$. (Taken from Selzer and Cirpka (2020).)

	$mean(\cdot)$	$median(\cdot)$	$\|\|(\|\cdot\|)\|\|_\infty$
$\boldsymbol{\epsilon}_{abs}$	1.05	1.04	0.45
$\boldsymbol{\epsilon}_{dir}$	$2.60 \cdot 10^{-2}$	$2.16 \cdot 10^{-2}$	$1.38 \cdot 10^{-1}$

Figures 7a and 7b show the corresponding particle trajectories for the FVM velocity field and the $\mathcal{RTN}_0$-projection of the $\mathcal{P}_1$ Galerkin FEM velocity field, respectively. Both trajectory patterns appear reasonable. The trajectories diverge in the recharge area depicted by the blue-gray

square. The trajectories of the FVM solution is again more angular, while the $\mathcal{RTN}_0$-projected solution preserves the smoothness of the $\mathcal{P}_1$ Galerkin FEM solution.

Like in the non-divergent case, we quantify the difference of the two conforming velocity fields in the divergent case depicted in Figures 7a and 7b by the ratio of absolute velocities according to Equation 118 and the angle between the two velocities according to Equation 119. Figures 7c and 7d show the spatial distributions of these two difference measures, and Table 2 lists norms over the entire domain. These metrics confirm that the two velocity fields are quite similar. All measures listed in Table 2 are very low. We also computed the infinity norm of the relative error in mass conservation, which was on the order of machine precision for double-precision real variables.

3.1.3 Empirical Consistency and Convergence Tests for Anisotropic Hydraulic Conductivity on Triangles

For evaluating the performance of the $\mathcal{RTN}_0$-projection in cases with anisotropic hydraulic conductivity, we consider a unit square $\Omega = [0,1]^2$ in which a fixed non-zero flux is defined as Neumann boundary condition over the central sections ($y \in [0.25, 0.75]$) of the left and right boundaries, all other boundary sections are no-flow boundaries. The domain is discretized by 812 elements and 439 nodes, the maximum diameter of the elements, $h_{\mathcal{T}}$, is chosen such that $1/h_{\mathcal{T}} = 16$. We consider a uniform hydraulic conductivity with the component $k_{xx} = 10^{-4}\,\mathrm{m/s}$ in x-direction remaining identical in all cases, whereas the k_{yy}-component in y-direction is varied such that $k_{yy}/k_{xx} = 1$, $k_{yy}/k_{xx} = 0.1$, and $k_{yy}/k_{xx} = 0.01$. The off-diagonal entries remain zero. The uniform porosity is $n_e = 0.4$. The normal flux at the inflow and outflow boundaries is uniformly set to $\mathbf{n} \cdot \mathbf{q}_{N,+} = -\mathbf{n} \cdot \mathbf{q}_{N,-} = -10^{-4}\,\mathrm{m/s}$. As in the preceding case, the starting positions of particles are chosen such that the spacing reflects an equal cumulative flux across the boundary, so that the particle trajectories represent streamlines. For illustration, see Figure 8, which illustrates the case of $k_{yy}/k_{xx} = 0.1$.

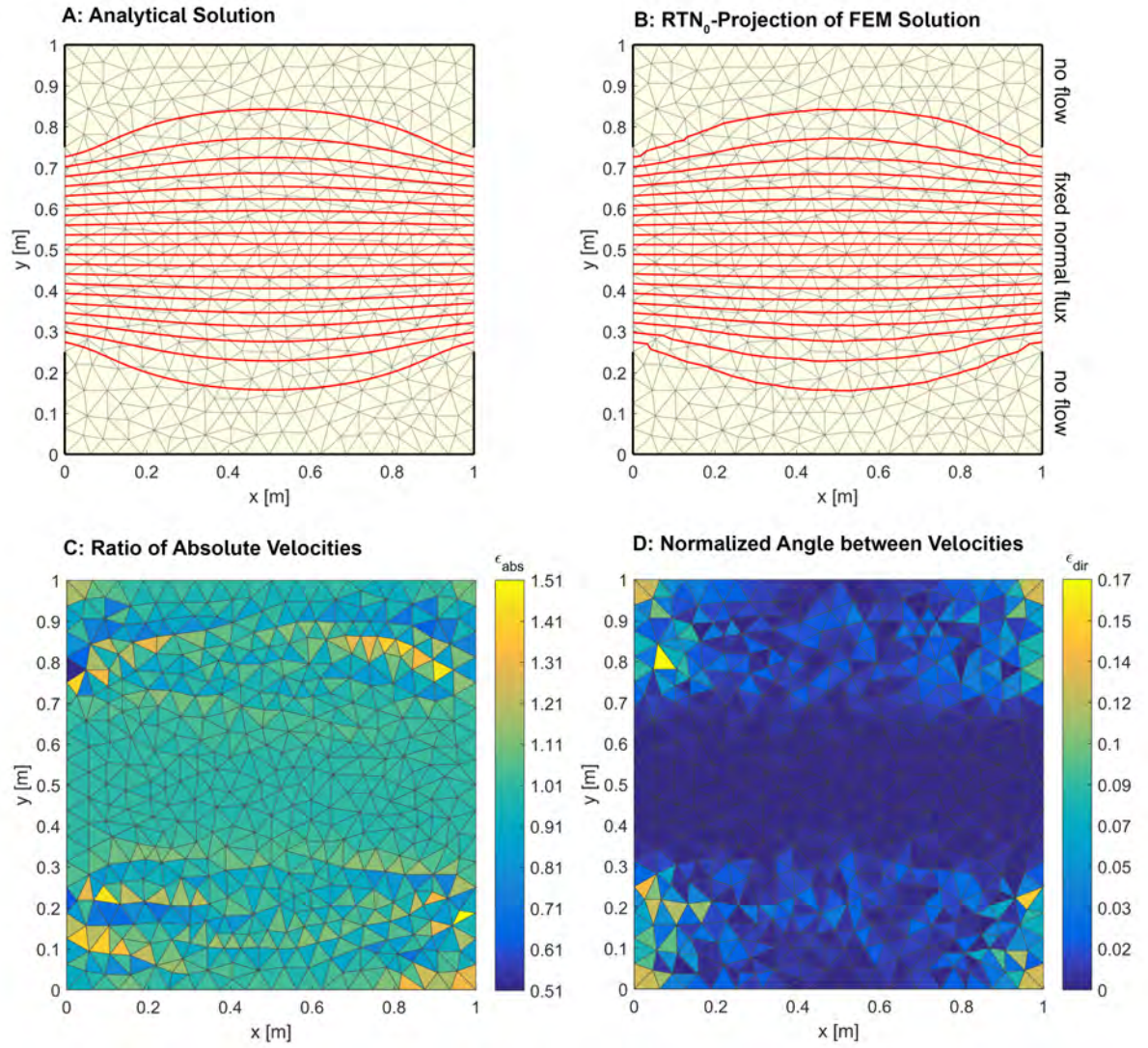

Figure 8: Non-divergent test case with hydraulic anisotropy for the case of $k_{yy}/k_{xx} = 0.1$. a) particle trajectories of the analytical solution (120); b) particle trajectories of the $\mathcal{RTN}_0$-projection of the $\mathcal{P}_1$ Galerkin FEM velocity field (bold black line at the boundary: no-flow boundary); c) ratio of absolute velocities at cell centroids ($\boldsymbol{\epsilon}_{abs}$); d) angle between the velocities at cell centroids ($\boldsymbol{\epsilon}_{dir}$). (This Figure is taken from Selzer and Cirpka (2020).)

We show consistency by comparing the results of the $\mathcal{RTN}_0$-projection to the corresponding analytical solution rather than a numerical result based on cell-centered FVM. The analytical solution for our test case is:

$$\mathbf{q}^{Ana} = \left(\left(\frac{Q}{L_y} - k_{xx} \sum_{r=1}^{\infty} \alpha_r \left(a_{\oplus,r} \exp\left(\alpha_r x\right) - a_{\ominus,r} \exp\left(-\alpha_r x\right) \right) \cos\left(\frac{2\pi y r}{L_y} \right) \right), \right.$$
$$\left. \left(k_{yy} \frac{2\pi}{L_y} \sum_{r=1}^{\infty} r \left(a_{\oplus,r} \exp\left(\alpha_r x\right) + a_{\ominus,r} \exp\left(-\alpha_r x\right) \right) \sin\left(\frac{2\pi y r}{L_y} \right) \right) \right) \tag{120}$$

where Q [$\mathrm{L}^2\mathrm{T}^{-1}$] is the total volumetric flux and L_y [L] denotes the total vertical length of the

domain, which equals unity in our case. The coefficients are given by:

$$a_{\oplus,r} = \frac{1 - \exp\left(-\alpha_r L_x\right)}{2\sinh\left(\alpha_r L_x\right)} \frac{\widetilde{q}_{in,r}}{k_{xx}\alpha_r}, \qquad \text{and} \qquad a_{\ominus,r} = \frac{1 - \exp\left(\alpha_r L_x\right)}{2\sinh\left(\alpha_r L_x\right)} \frac{\widetilde{q}_{in,r}}{k_{xx}\alpha_r}, \tag{121}$$

and

$$\widetilde{q}_{in,r} = \frac{2}{r\pi} \frac{Q}{L_w} \sin\left(\frac{r\pi\left(L_y - L_w\right)}{L_y}\right), \qquad \text{and} \qquad \alpha_r = \frac{2\pi r}{L_y}\sqrt{\frac{k_{yy}}{k_{xx}}}, \tag{122}$$

where L_w [L] is the length of the inflow/outflow boundary and L_x [L] denotes the total horizontal length of the domain, which equals unity in our case.

Table 3: Measures for the differences between the $\mathcal{RTN}_0$-projection in 2-D and the analytical solution for three anisotropy ratios k_{yy}/k_{xx}. As $\boldsymbol{\epsilon}_{abs}$ is a ratio, we give the respective infinity norm as $||(|1 - \boldsymbol{\epsilon}_{abs}|)||_\infty$. (Taken from Selzer and Cirpka (2020).)

k_{yy}/k_{xx}		$mean(\cdot)$	$median(\cdot)$	$\|\|(\|\cdot\|)\|\|_\infty$
1	$\boldsymbol{\epsilon}_{abs}$	1.00	1.00	0.31
	$\boldsymbol{\epsilon}_{dir}$	$1.04 \cdot 10^{-2}$	$4.16 \cdot 10^{-3}$	$1.30 \cdot 10^{-1}$
0.1	$\boldsymbol{\epsilon}_{abs}$	1.00	1.00	0.51
	$\boldsymbol{\epsilon}_{dir}$	$1.50 \cdot 10^{-2}$	$5.80 \cdot 10^{-3}$	$1.72 \cdot 10^{-1}$
0.01	$\boldsymbol{\epsilon}_{abs}$	3.12	1.02	23.73
	$\boldsymbol{\epsilon}_{dir}$	$2.64 \cdot 10^{-1}$	$1.72 \cdot 10^{-2}$	1.00

Figures 8a and 8b show the corresponding particle trajectories based on the analytical solution and the $\mathcal{RTN}_0$-projection of the $\mathcal{P}_1$ Galerkin FEM velocity field, respectively, for the case of an anisotropy ratio of 0.1. Figures 8c and 8d show the ratio of absolute velocities of the $\mathcal{RTN}_0$-projection over the analytical solution and the difference in the direction, respectively. The trajectory pattern gives the visual impression that the result of the $\mathcal{RTN}_0$-projection of the $\mathcal{P}_1$ Galerkin FEM velocity field is very close to the analytical solution. The patterns of the two other anisotropy ratios show comparable similarities, but the trajectories are more centered in the middle section for $k_{yy}/k_{xx} = 0.01$ and more evenly distributed in the vertical direction for $k_{yy}/k_{xx} = 1$.

Table 3 lists the norms of difference measures, confirming the good agreement of the $\mathcal{RTN}_0$-projection with the analytical solution for the isotropic case and for the case with the anistropy ratio of $k_{yy}/k_{xx} = 0.1$. For an anisotropy ratio of $k_{yy}/k_{xx} = 0.01$, the $\mathcal{RTN}_0$-projection of the $\mathcal{P}_1$ Galerkin FEM velocity introduces numerical artifacts, which mainly occur in low flow regions near the no-flow boundaries. Already the case of $k_{yy}/k_{xx} = 0.1$, shown in Figures 8c and 8d indicates these tendencies. For $k_{yy}/k_{xx} = 0.01$, however, the orientation of the velocities near the top and the bottom is mostly reverted. In these regions the absolute velocity is significantly smaller than in the central section, particularly in cases of high anisotropy. Because of the small values, getting the velocity right in these regions has a relative low significance in the minimization procedure, which induces the described numerical artifacts.

To analyze the performance of the $\mathcal{RTN}_0$-projection more rigorously, we extend this test by an empirical convergence analysis comparing our obtained results with those of the base $\mathcal{P}_1$ Galerkin FEM solution. We employ the same model as exemplarily shown in Figure 8 using iteratively refined grids. The base grid discussed above has a maximum diameter of the elements, $h_\mathcal{T}$, such that $1/h_\mathcal{T} = 16$. For the empirical convergence analysis we use this grid amongst other refinement levels, whereas keeping all adjustments for the grid generation identical on all levels.

Table 4: Convergence behavior of the numerically approximated velocity fields for different anisotropy ratios k_{yy}/k_{xx} of hydraulic conductivity with $k_{xx} = 1 \cdot 10^{-4}$ m/s. The empirical order of convergence is given in parentheses. $h_\mathcal{T}$ is the maximum diameter of the triangles, i.e., the maximum face length, and N is the number of elements. Furthermore, $\mathbf{q}^{RTN_0}$ and $\mathbf{q}^{Ana}$ are the velocity fields of the $\mathcal{RTN}_0$-projection and the analytical solution, respectively. (Adapted from Selzer and Cirpka (2020).)

k_{yy}/k_{xx}	$1/h_T$	N	$\frac{1}{N}\sum_{E_i\in\mathcal{T}} \|\|\mathbf{q}_{E_i}^{RTN_0}(\mathbf{x}_c) - \mathbf{q}_{E_i}^{Ana}(\mathbf{x}_c)\|\|_2$	$\frac{1}{N}\sum_{E_i\in\mathcal{T}} \|\|\mathbf{q}_{E_i}^{FE}(\mathbf{x}_c) - \mathbf{q}_{E_i}^{Ana}(\mathbf{x}_c)\|\|_2$
1	4	48	$2.15 \cdot 10^{-5}$ (−)	$2.14 \cdot 10^{-5}$ (−)
	8	196	$1.07 \cdot 10^{-5}$ (1.01)	$1.09 \cdot 10^{-5}$ (0.97)
	16	812	$5.36 \cdot 10^{-6}$ (1.00)	$5.45 \cdot 10^{-6}$ (1.00)
	32	3354	$2.68 \cdot 10^{-6}$ (1.02)	$2.73 \cdot 10^{-6}$ (1.00)
0.1	4	48	$2.21 \cdot 10^{-5}$ (−)	$4.04 \cdot 10^{-5}$ (−)
	8	196	$1.16 \cdot 10^{-5}$ (0.93)	$1.74 \cdot 10^{-5}$ (1.22)
	16	812	$6.31 \cdot 10^{-6}$ (0.88)	$8.72 \cdot 10^{-6}$ (0.99)
	32	3354	$3.16 \cdot 10^{-6}$ (0.99)	$4.38 \cdot 10^{-6}$ (0.99)
0.01	4	48	$2.23 \cdot 10^{-5}$ (−)	$5.41 \cdot 10^{-5}$ (−)
	8	196	$1.97 \cdot 10^{-5}$ (0.18)	$2.77 \cdot 10^{-5}$ (0.97)
	16	812	$1.02 \cdot 10^{-5}$ (0.95)	$1.61 \cdot 10^{-5}$ (0.78)
	32	3354	$4.48 \cdot 10^{-6}$ (1.19)	$8.19 \cdot 10^{-6}$ (0.97)

According to Table 4 the velocity field approximated by the $\mathcal{RTN}_0$-projection essentially shows the same order of convergence as the original velocity field based on the $\mathcal{P}_1$ Galerkin FEM. It is also worth noting that, besides of the coarsest grid for an anisotropy ratio of unity, all other norms indicate that the conforming velocity field of the $\mathcal{RTN}_0$-projection is overall closer to the analytical solution than the original flux approximation of the $\mathcal{P}_1$ Galerkin FEM. Nevertheless, the numerical artifacts of reverted velocity fields and wrong magnitudes of the velocities induced by the $\mathcal{RTN}_0$-projection for an anisotropy ratio of $k_{yy}/k_{xx} = 0.01$ are also preserved in the reduced empirical convergence order shown in the first refinement step in Table 4.

We conclude the test case by reporting on the computation time needed to solve the equation systems resulting from the $\mathcal{RTN}_0$-projection for the isotropic case, for which we can also perform the cell-centered FVM simulations as a comparison. The systems of equations are solved by the direct solver UMFPACK (Davis, 2004), accessed via the backslash-operator of Matlab.

Table 5: Comparison of the computational costs for the $\mathcal{RTN}_0$-projection and the cell-centered FVM for the isotropic case given in CPU-time and wall time in seconds needed to solve the final systems of equations given as median value over 11 runs each. $h_\mathcal{T}$ is the maximum diameter of the triangles, N is the number of elements, and M is the number of element edges. (Adapted from Selzer and Cirpka (2020).)

$1/h_T$	N	M	CPU-time (wall time): $\mathcal{RTN}_0$-projection	CPU-time (wall time): cell-centered FVM
4	48	68	0.09375 (0.054655)	0.09375 (0.022623)
8	196	286	0.09375 (0.038496)	0.1250 (0.044168)
16	812	1202	0.296875 (0.062972)	0.203125 (0.053037)
32	3354	4999	0.265625 (0.069596)	0.046875 (0.01447)

The CPU-times shown in Table 5 clearly indicate that the final system of equations resulting from the cell-centered FVM can be solved quicker for larger systems than the system resulting from the $\mathcal{RTN}_0$-projection. This is so because the system of equations resulting from the cell-centered FVM discretization is symmetric, positive-definite, and of order N, whereas the system of equations resulting from the $\mathcal{RTN}_0$-projection is, though symmetric, an indefinite saddle-point problem of order $N + M$ comparable to an $\mathcal{RTN}_0$ MFEM discretization.

3.1.4 Three-Dimensional Test Case

Our three-dimensional benchmark model is a cube with an edge length of 10.0 m. The background isotropic hydraulic conductivity is $k = 10^{-4}$ m/s. Two cuboid inclusions ($x \in [2, 4]$, $y \in [0, 5]$, $z \in [2, 4]$ and $x \in [6, 8]$, $y \in [5, 10]$, $z \in [6, 8]$, respectively) have a strongly reduced isotropic hydraulic conductivity of $k = 10^{-10}$ m/s. In Figure 9a, the inclusions are highlighted as blue blocks. The uniform porosity is $n_e = 0.4$. The domain is discretized into cubes of 1 m edge length, which are split into six tetrahedra such that effects of grid orientation in the cell-centered FVM solution are minimized. This leads to 6000 tetrahedral elements and 1331 nodes. The left boundary of the domain (i.e., the set of faces with $x = 0$ m) and the right boundary (i.e., the set of faces with $x = 10$ m) are Dirichlet boundaries with uniform hydraulic-head values, in which the head difference is 1.0 m between the left and right boundaries. No flow is permitted across all other boundaries of the domain.

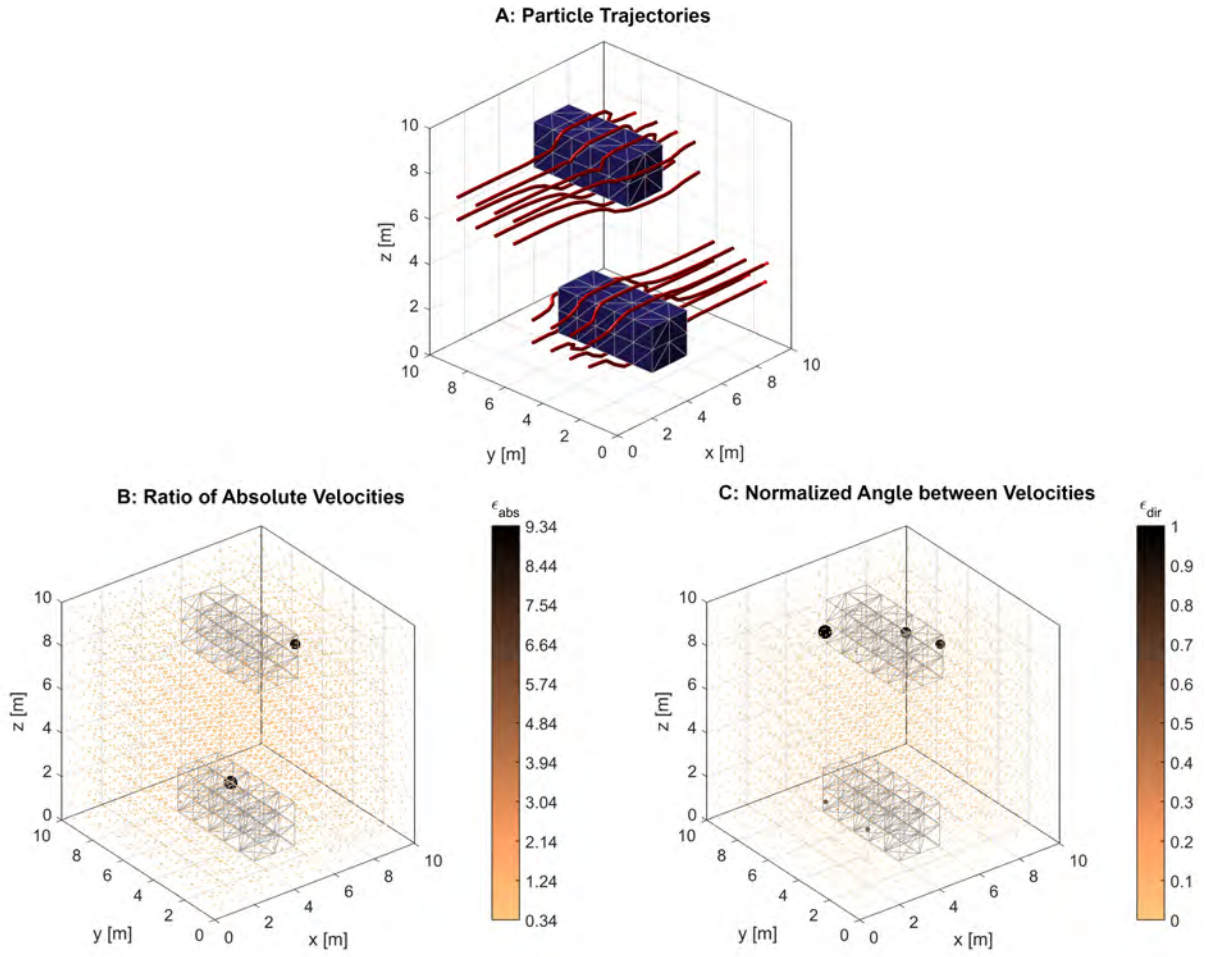

Figure 9: Three-dimensional, non-divergent test case with two nearly impermeable cuboid inclusion. a) particle tracks based on the proposed $\mathcal{RTN}_0$-projection of the $\mathcal{P}_1$ Galerkin FEM velocity approximation on tetrahedra; b) ratio of the absolute velocity resulting from the $\mathcal{RTN}_0$-projection and that from the cell-centered FVM solution at the cell centroids ($\boldsymbol{\epsilon}_{abs}$); c) angle between the two velocity approximations at the cell centroids ($\boldsymbol{\epsilon}_{dir}$). (Taken from Selzer and Cirpka (2020).)

Figure 9a shows a few particle trajectories starting at the inlet boundary of the domain, which have been computed using the $\mathcal{RTN}_0$-projection of the specific-discharge field originating from the $\mathcal{P}_1$ Galerkin FEM solution. The particle trajectories circumvent the inclusions in a physically reasonable manner. In contrast to the two-dimensional test cases, we don't show particle trajectories for the FVM solution, because the visual impression does not indicate any obvious differences to trajectories obtained from the $\mathcal{RTN}_0$-projected velocity field of the standard FEM solution.

Table 6: Measures for the differences between the discretization methods in the 3-D application with two nearly impervious cuboid inclusions. As $\boldsymbol{\epsilon}_{abs}$ is a ratio, we give the respective infinity norm as $||(|1-\boldsymbol{\epsilon}_{abs}|)||_\infty$. (Taken from Selzer and Cirpka (2020).)

	$mean(\cdot)$	$median(\cdot)$	$\|\|(\|\cdot\|)\|\|_\infty$
$\boldsymbol{\epsilon}_{abs}$	$9.89 \cdot 10^{-1}$	$9.74 \cdot 10^{-1}$	8.34
$\boldsymbol{\epsilon}_{dir}$	$5.83 \cdot 10^{-2}$	$5.86 \cdot 10^{-2}$	1

Figures 9b and 9c visualize the measures of differences in the velocity field between the FVM solution and the velocity approximation by our $\mathcal{RTN}_0$-projection. Figure 9b shows the ratio of absolute velocities, and Figure 9c the normalized angle between the two velocity vectors at the cell centroids. Table 6 lists the associated norms for the entire domain. The magnitude of the differences are visualized in Figures 9b and 9c by both the size and color of spheres at the cell centroids. In the largest part of the domain, these differences are negligible. Only at certain points next to the inclusions the difference measures exceed average values. The difference in the direction is mainly caused by a slight shift of points where the particle trajectories separate to circumvent the inclusions. By coincidence, two cells show a larger difference in the absolute value of the velocity.

As indicated by the difference metrics listed in Table 6, the results are very similar. Like in the 2-D test cases, the trajectories based on the FVM solution on simplices are more angular than those based on the $\mathcal{RTN}_0$-projection of the $\mathcal{P}_1$ Galerkin FEM solution, because the projection tries to preserve the smoothness of the standard FEM solution. We would deem the projected results to be physically more accurate, but the FVM solution is achieved at considerably lower computational costs.

Optimal splitting of structured grids of cubes into conforming grids of tetrahedra

For the example computations depicted in Figure 9, we split a regular grid of cubes into a conforming grid of tetrahedra such that the effects of grid orientation in a lowest-order FVM solution are minimized. In this excursus, we develop our splitting strategy, naming and depicting also other strategies in order to show that our splitting approach is superior.

There are infinite ways how one can split a three-dimensional polyhedron into tetrahedra because, once split into tetrahedra, every tetrahedron can be split again in tetrahedra. Possibilities include cutting the tetrahedron in halfs starting from one vertex introducing an additional face spanning from the according vertex to the median of the triangular base area opposite to the vertex. By this, one tetrahedron is split into two. Another common approach is the addition of an additional internal vertex, e.g., at the centroid of the tetrahedron (see e.g., Dompierre et al., 1999). By this, one tetrahedron can be split into four tetrahedra. However, in the following we focus on schemes which do not introduce additional vertices, which reduces the possibilities to split a cube into tetrahedra to a finite amount. The minimum amount of tetrahedra in which a cube can be split is five (e.g, Cordes and Kinzelbach, 1992; Dompierre et al., 1999), where one tetrahedron is in the middle of the cube, being surrounded by four tetrahedra which faces are, among others, part of the former faces of the cube. However, in this section, we opt to split the cubes first into two triangular prisms, followed by a division of every triangular prism into three tetrahedra. Three tetrahedra is the minimum amount in which a triangular prism can be split. The reason for splitting a cube into six tetrahedra is, that this approach can lead to a possible configuration how a conforming grid of tetrahedra can be defined such that the effects of grid orientation introduced especially by a lowest-order FVM or MFEM are minimized (Selzer and Cirpka, 2020; Putti and Sartoretto, 2009; Hoteit et al., 2002). Moreover, the cubes are not only embedding tetrahedra but also triangular prisms, this makes computations on different

geometries possible using the same spatial base discretization.

In our exemplification, we consider a grid of 1000 unit cubes, which equals the base discretization of the results depicted in Figure 9 spanned such that the resulting domain has the dimensions $10\,\mathrm{m} \times 10\,\mathrm{m} \times 10\,\mathrm{m}$. For our particle tracking examples the same boundary conditions as in the three-dimensional test case of this section are chosen. The only difference is that now a uniform, isotropic hydraulic conductivity of $k = 10^{-4}\,\mathrm{m/s}$ is assumed everywhere such that flow should be perfectly straight from the inlet to the outlet of the cuboid, which equals in the projection from the top on the base of the domain flow-paths which should all go from left to right. For the particle tracking examples, we consider 96 particles which are released on the Dirichlet boundary on the left, which is the inflow boundary. The starting locations of the particles are regularly spaced in y and z with $x = 0$, where $\mathbf{x} = (x, y, z)$, which yields a quadrilateral pattern of starting locations projected on the inflow boundary around the centroid of the inflow boundary being the center. Therefore, the mean of all starting particle locations equals $(0, 5, 5)$. For the approximation of the overall drift, we will give the mean coordinates of the particles at their exit locations. Note that, according to the model setup, in the ideal case the mean of the starting locations should equal the mean of the exiting locations in y and z while x should equal 10, leading to the ideal mean of exit coordinates of $(10, 5, 5)$. Note also that in the following only conforming grids of tetrahedra are discussed.

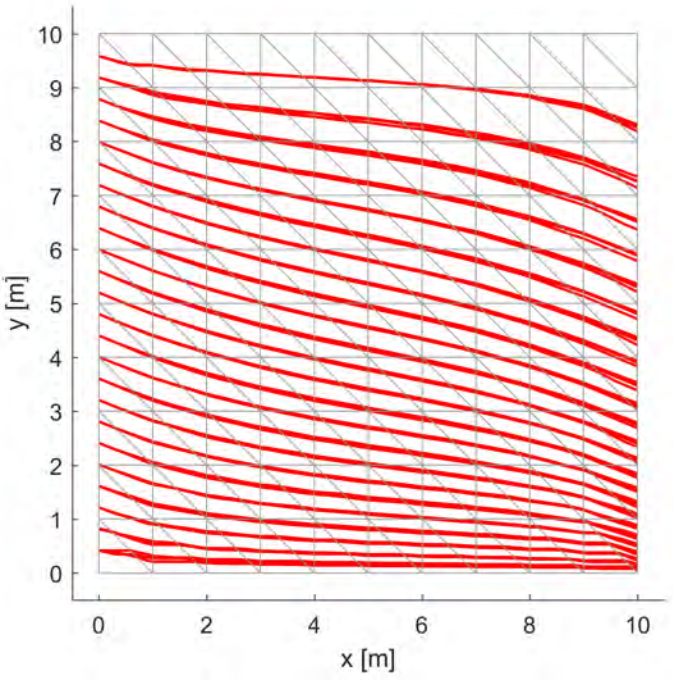

Figure 10: Naive split of a grid of structured cubes first split into triangular prisms and in a second step into a grid of conforming tetrahedra following a consistent, non-alterating strategy. View from the top on 96 particle trajectories in different locations in y and different depths in z.

In Figure 10 a naive split of structured unit cubes into a conforming grid of tetrahedra using a consistent, non-alterating strategy is depicted seen from the top. Every cube is first split

into two prisms always by inserting an additional face from the top left to the bottom right in the current projection. Then, every prism is split into three tetrahedra. Particle trajectories are dipping not only from high values in y to low values in y, like it can be seen in Figure 10, but they are also dipping form their initial vertical coordinates in z to lower values in z. The mean of the exit coordinates for the splitting strategy depicted in Figure 10 is rounded to $(10, 2.86, 2.47)$ which is considerably different from the ideal mean of the exit coordinates, which is $(10, 5, 5)$. The deviation is about a factor of two, which is a severe difference significantly influencing the computed trajectories. The deviation can solely be explained by the effects of grid orientation induced by a suboptimal splitting procedure combined with the lowest-order FVM approximation described in section 2.4.3 .

In the following we explain different strategies of splitting grids of structured cubes into grids of conforming tetrahedra, which are better, finally developing our own strategy, where the effects of grid orientation are averaged out such that the final exit coordinates of all particles equal their ideal exit coordinates being the analytical solution up to numerical accuracy. First, we introduce our numbering strategy of nodes, cubes, and triangular prisms.

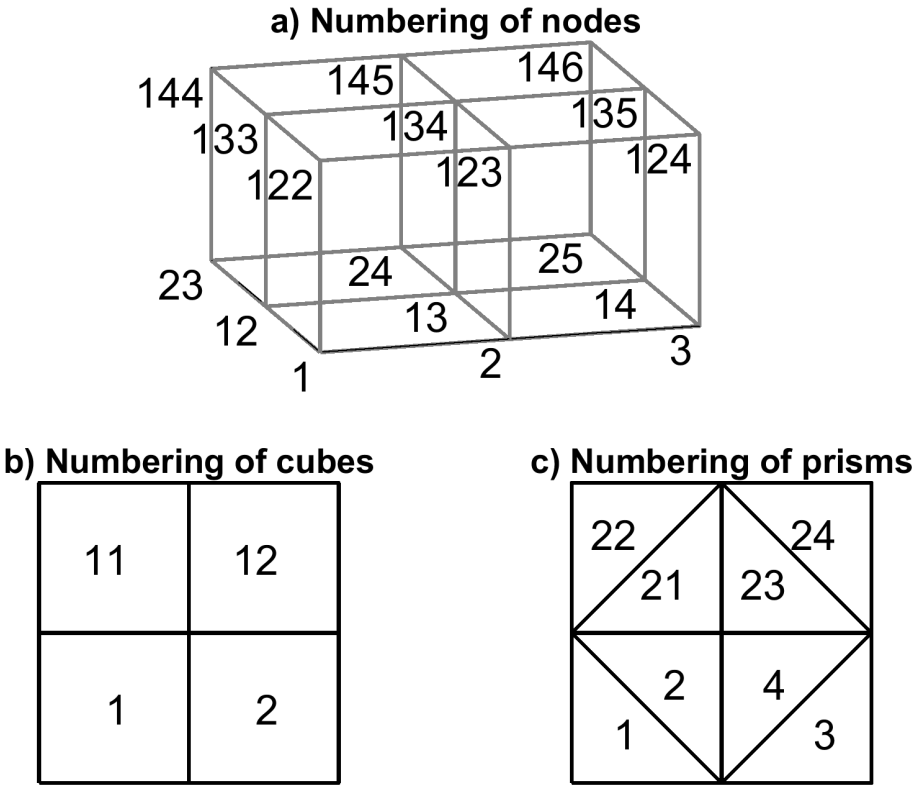

Figure 11: Numbering strategies for nodes, cubes, and prisms exemplified for the four cubes being in the bottom left of the domain next to the origin of the coordinate system. In general, numbering is from left to right and from bottom to top. a) Numbering of nodes; b) numbering of cubes seen from the top; c) numbering of triangular prisms seen from the top. Note that the whole domain is split into 4-tuples of cubes, where every cube only belongs to one tuple.

In Figure 11a the numbering strategies of nodes is exemplified for the four cubes of the primal grid next to the origin in the bottom left of the domain next to the origin of the coordinate system. In our example grid, every horizontal plane has 121 nodes, and there are 11 horizontal nodal planes. Figure 11b depicts the numbering strategy of cubes from low to high values in x, in y, and also in z, which is not depicted in the Figure. Figure 11c exemplifies the numbering strategy of triangular prisms for the four cubes next to the origin of the coordinate system. The numbering strategy of the four cubes as well as of the triangular prisms is continued accordingly for all other 4-tuples of cubes and derived triangular prisms as a series with increasing values in x, y, and z.

For the splitting, we define the example cubes and prisms as node-tuples consisting of all nodes in ascending order of node IDs. Considering the numbering strategy depicted in Figure 11 the four cubes are defined as node-tuples: cube 1 = $(1, 2, 12, 13, 122, 123, 133, 134)$, cube 2 = $(2, 3, 13, 14, 123, 124, 134, 135)$, cube 11 = $(12, 13, 23, 24, 133, 134, 144, 145)$, and cube 12 = $(13, 14, 24, 25, 134, 135, 145, 146)$. Following again the ascending order of node IDs, the node-tuples defining the exemplary triangular prisms are: prism 1 = $(1, 2, 12, 122, 123, 133)$, prism 2 = $(2, 12, 13, 123, 133, 134)$, prism 3 = $(2, 3, 14, 123, 134, 135)$, prism 4 = $(2, 13, 14, 123, 134, 135)$, prism 21 = $(12, 13, 24, 133, 134, 145)$, prism 22 = $(12, 23, 24, 133, 144, 145)$, prism 23 = $(13, 14, 24, 134, 135, 145)$, prism 24 = $(14, 24, 25, 135, 145, 146)$. For splitting triangular prisms into tetrahedra, the reader is also referred to Dompierre et al. (1999), who do not only describe individual splitting strategies for single pyramids, triangular prisms, and hexahedra, but also an algorithm for splitting grids of unstructured triangular prisms in grids of conforming tetrahedra. In our tuples, defining the triangular prisms, we consider the ordinal numbers, which are the relative positions of the node IDs in a tuple, which implies that they range from one to six for triangular prisms. Considering these positions, the six variants how a triangular prism can be split into tetrahedra are the following listed in Table 7.

Table 7: Strategies of splitting triangular prisms in tetrahedra. The tetrahedra are given as tuples consisting of ordinal numbers referring to the relative position of a node ID in the tuples of triangular prisms defined as above.

	Tetrahedron 1	Tetrahedron 2	Tetrahedron 3
Strategy 1	$(1, 2, 3, 6)$	$(1, 2, 5, 6)$	$(1, 4, 5, 6)$
Strategy 2	$(1, 2, 3, 5)$	$(1, 3, 4, 5)$	$(3, 4, 5, 6)$
Strategy 3	$(1, 2, 3, 5)$	$(1, 3, 5, 6)$	$(1, 4, 5, 6)$
Strategy 4	$(1, 2, 3, 4)$	$(2, 3, 4, 6)$	$(2, 4, 5, 6)$
Strategy 5	$(1, 2, 3, 6)$	$(1, 2, 4, 6)$	$(2, 4, 5, 6)$
Strategy 6	$(1, 2, 3, 4)$	$(2, 3, 4, 5)$	$(3, 4, 5, 6)$

Using the splitting strategies shown in Table 7, we depict four possible strategies of splitting triangular prisms, derived from cubes, into conforming grids of tetrahedra.

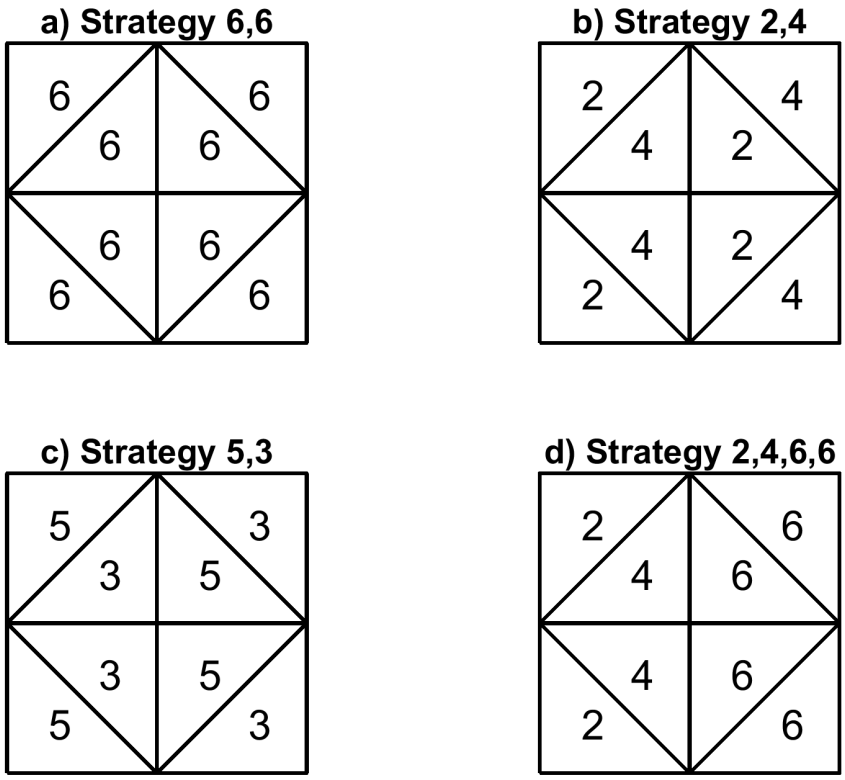

Figure 12: Splitting strategies of cubes seen from the top as a two-dimensional projection first into triangular prisms denoted by the diagonal lines and second into tetrahedra exemplified for a 4-tuple of cubes. The numbers within the triangular prisms, which are triangles seen from the top, denote the splitting strategy for the individual prism according to the strategies listed in Table 7. The headings of the sub-figures name the overall splitting strategy for a 4-tuple of cubes, and therefore, for the domain, as the splitting is repeated accordingly for all 4-tuples in the domain. All splitting strategies shown lead to conforming grids of tetrahedra. a) Splitting strategy 6,6; b) splitting strategy 2,4; c) splitting strategy 5,3; d) splitting strategy 2,4,6,6 merging the strategies 2,4 and 6,6.

In Figure 12a-d four different strategies for splitting cubes into grids of conforming tetrahedra are depicted and named via the headings for a 4-tuple of cubes. Note that the same splitting is repeated for all other 4-tuples of cubes such that the whole domain is covered. All four strategies shown in Figure 12a-d lead to conforming grids of tetrahedra. However, although they are all better than the splitting procedure employed for Figure 10, they may still induce grid effects of different magnitude and orientation when employed in a lowest-order FVM or MFEM discretization.

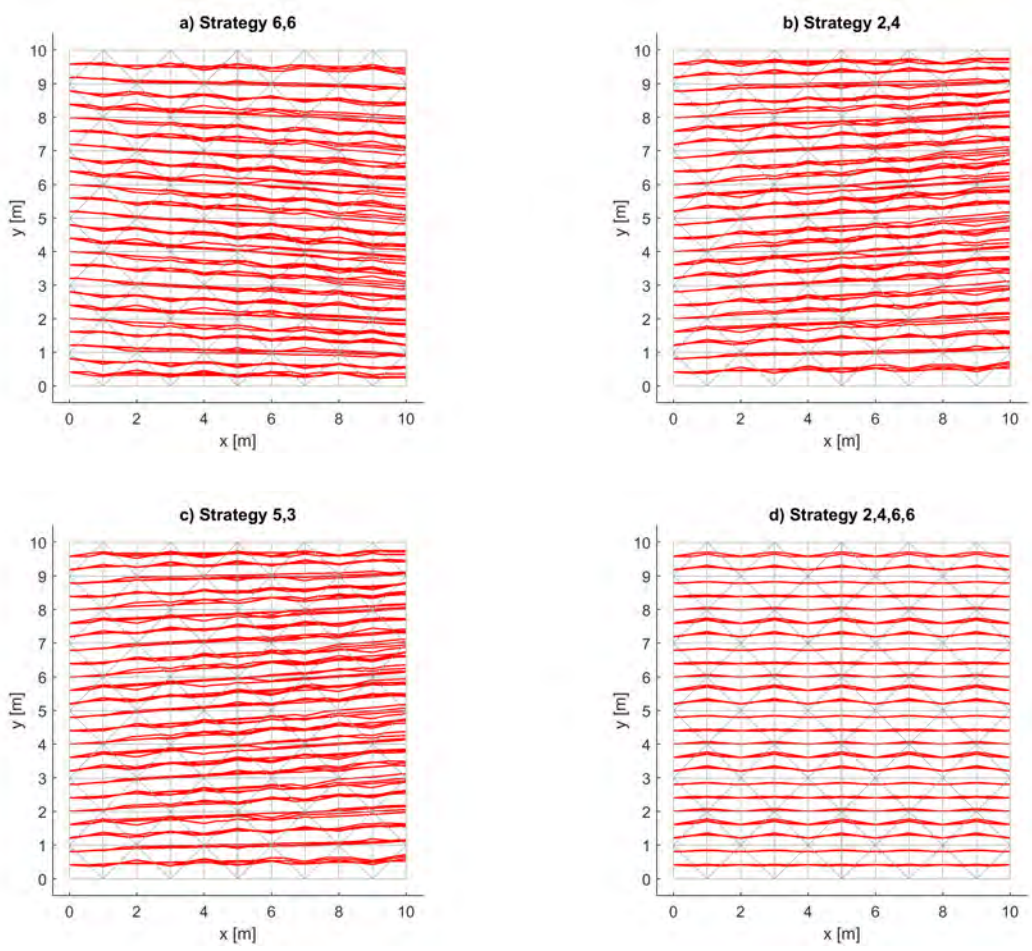

Figure 13: Trajectories for 96 particle released in different locations with $x = 0$ with regular spacing in y and z for the different splitting strategies of cubes into conforming grids of tetrahedra according to the strategies depicted in Figure 12a-d in a cuboid seen from the top. a) Particle trajectories for strategy 6,6; b) particle trajectories for strategy 2,4; c) particle trajectories for strategy 5,3; d) particle trajectories for strategy 2,4,6,6.

In Figure 13a-d particle trajectories are shown which are associated to the four different splitting strategies depicted in Figure 12a-d. The numbering and the titles are the same for different subfigures indicating which trajectory pattern belongs to which splitting strategy. Figure 13a shows the results for the first splitting strategy depicted in Figure 12a, where all triangular prisms are split according to strategy 6,6 (see Table 7 for this strategy and for all other strategies). The mean of the exit coordinates for strategy 6,6 is $(10, 4.49, 4.02)$ which is closer to the ideal exit coordinates, which are $(10, 5, 5)$ and which equal the analytical solution, than the results for the naive split (see Figure 10). Still, the exit coordinates obtained with the splitting strategy 6,6 differ from the analytical solution. The trajectories deviate towards too low values in y and z. This is an effect solely due to the grid orientation affecting the lowest-order FVM approximation of groundwater flow, the same holds for grid effects observed in all other splitting strategies. In Figure 13b the results for the strategy 2,4 (compare Figure 12b) are depicted. Here, the opposite effect, compared to the results of strategy 6,6 can be observed. The mean of the exit coordinates for the strategy 2,4 is $(10, 5.51, 5.98)$. Particles deviate towards too high values in

y and z. Figure 13c displays the results for the strategy 5,3 (compare 12c). Again, grid effects can be observed. The mean of the exit coordinates for strategy 5,3 is $(10, 5.51, 4.02)$, for this strategy the trajectories deviate towards too high values in y and too low values in z. Finally, Figure 13d shows the results for the strategy 2,4,6,6 (compare 12d), which is a merge of the strategies 2,4 (Figure 13b) and 6,6 (Figure 13a). Note, that these strategies are merged such that the resulting grid of tetrahedra obtained by strategy 2,4,6,6 is still conforming. Many other combinations of the individual splitting strategies displayed in the Figures 13a-c do not yield conforming grids. However, the mean of the exit coordinates of the splitting strategy 2,4,6,6, is $(10, 5, 5)$. All exit coordinates of the individual particle trajectories equal the analytical solution up to numerical precision. Therefore, the splitting strategy 2,4,6,6 cannot only be considered superior but optimal, it is used for the example computations of this chapter depicted in Figure 9. Although the exact particle trajectories in the domain differ from the analytical solution, the effects of grid orientation are averaged out for the particle trajectories due to the splitting procedure such that the exit coordinates of the particle trajectories again equal the analytical solutions.

3.1.5 The $\mathcal{RTN}_0$-Projection for Variably Saturated Flow – An extension on Triangular Prisms

In this section, we describe the results for the extension of the $\mathcal{RTN}_0$-projection to variably saturated flow. In the non-linear Richards' equation flow-determining parameters, like relative permeability, are themselves a function of the primary unknown, and very heterogeneous in space and time. For our application of the $\mathcal{RTN}_0$-projection to variably saturated flow, we replace the former weights that were the inverse of the hydraulic-conductivity tensor, $\mathbf{K}_{E_i}$, at the centroids of the elements (see section 2.5.1 and especially Equation 54), by the inverse of the variably saturated hydraulic-conductivity tensor, $k_{rel,c_i}\mathbf{K}_{E_i}$, where k_{rel,c_i} is the relative permeability associated to the centroid, c_i, of element E_i. However, in a finite-element-type discretization, the relative permeability is a nodal property and a function of the pressure head. Moreover, several relative permeability values may be associated to the nodes, depending on which element sharing the node is considered, if unsaturated material properties differ in the adjacent elements and flow is unsaturated. For an extension to variably saturated flow, we therefore need to interpolate the relative permeability to the element centroids. We do this in the following test case by linear interpolation of the nodal relative permeabilities to the centroid for each element, this procedure is equal to taking the arithmetic mean of the nodal relative permeabilities. Note furthermore, that this test case is not computed on simplices, but on non-deformed triangular prisms. In order to facilitate a solution on deformed triangular prisms, too, and for complying with the standard finite-element procedure, the extension of the $\mathcal{RTN}_0$-projection on triangular prisms for variably saturated flow is defined in local coordinates using a fixed reference prism and contravariant Piola transformation (compare e.g., Rognes et al. (2009)). Our reference prism is a non-standard one, it has a unit height, and an equilateral triangle as base with a base area of 0.5, its coordinates are $\mathbf{X}_{pri,ref} = ((0,0,0),(1/0.75^{1/4},0,0),((1/0.75^{1/4})/2,0.75^{1/4},0),(0,0,1),(1/0.75^{1/4},0,1),((1/0.75^{1/4})/2,0.75^{1/4},1))$. Using a reference prism with an equilateral base area

is beneficial for a velocity projection avoiding bookkeeping on which local face of the reference prism, a face in global coordinates is projected onto. The base area of 0.5 combined with a unit height simplify the shape functions describing the triangular prism. These definitions are set for convenience, other definitions are possible, too.

The described procedure uses a non-conforming, approximate primal velocity vector evaluated at the element centroid for minimizing the difference between a non-conforming velocity field and a conforming one in $\mathcal{RTN}_0$-space. However, this is only an approximation including several simplifications and assumptions. First, even for flow without sources or sinks, in opposite to simplicial elements, the velocity field is in general not constant anymore on non-simplicial elements neither for saturated flow nor for variably saturated flow due to the bi-linear, or multi-linear hydraulic-head approximation in the element, respectively. The fact, that relative permeability is defined on the nodes, and can even differ on the same node for adjacent elements complifies the issue further. Moreover, a standard Galerkin approximation of variably saturated flow in more than one spatial dimension is known to be prone to instabilities, as a fully consistent upstream-weighting procedure of relative permeability within the elements, is not feasible in standard finite elements in more than one spatial dimension (compare sections 2.4.1 and 2.4.2). Therefore, a finite-element-type primal solution is obtained using HydroGeoSphere in finite-difference mode, which is a finite difference method defined in finite-element terms, and mass lumping (compare section 2.4.2). However, in such a finite-difference approximation in finite-element terms, the solution is only defined on lines between the nodes, and not at the centroids, where a solution is needed to approximate a conforming velocity field via an $\mathcal{RTN}_0$-projection. That is, throughout this section, we use the velocity output of HydroGeoSphere, which is a velocity vector at the centroids approximated using the (full) finite-element shape functions and the nodal hydraulic-head solution evaluated at the centroid as well as the material properties, which are hydraulic conductivity and relative permeability, although we employ the described finite-difference approach in finite-element terms for computing the primal solution.

For our test case, we consider a benchmark model, which is a cubic domain having dimensions of $10\,\mathrm{m} \times 10\,\mathrm{m} \times 10\,\mathrm{m}$. The domain is discretized by 12,692 nodes and 22,572 elements in 18 elemental layers, the vertical discretization is refined at the top of the domain and around the assigned height of the Dirichlet boundary conditions. On the left and right sides of the domain we assign homogeneous Dirichlet boundary conditions with a constant hydraulic head $h_{fix} = 6\,\mathrm{m}$. The front, back, and bottom sides are no-flow boundaries. A recharge zone is located at the center of the top of the domain. The zone is a square with dimensions of $3\,\mathrm{m} \times 3\,\mathrm{m}$, and the applied recharge rate is $200\,\mathrm{mm/a}$. We assume a uniform saturated hydraulic conductivity of $10^{-5}\,\mathrm{m/s}$, saturated and residual water contents of 0.4 and 0.065, respectively, and van Genuchten parameters $\alpha = 7.5\,1/\mathrm{m}$ and $N = 1.89$ being all constant in the domain.

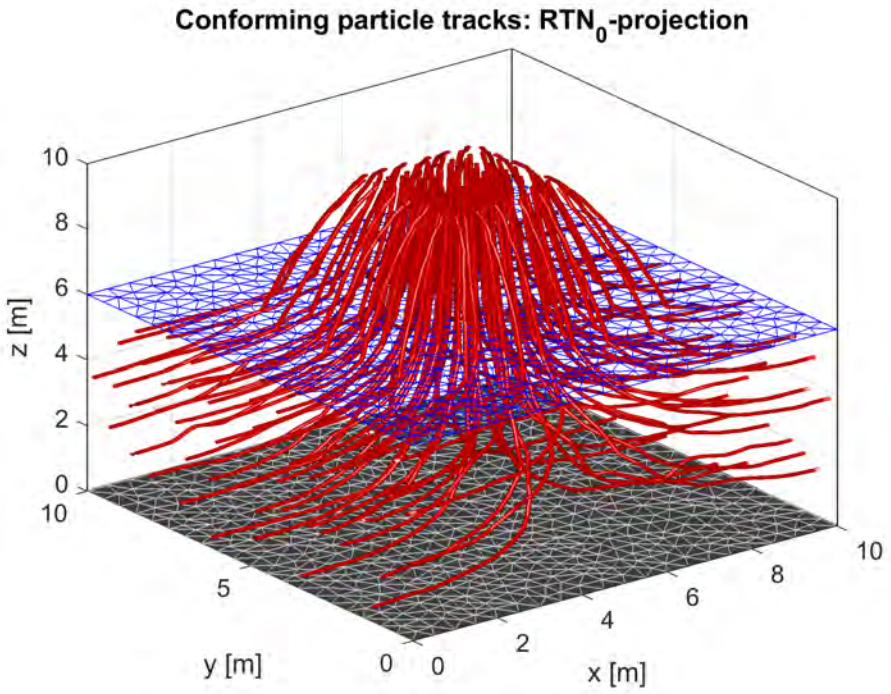

Figure 14: Benchmark problem with recharge via a square. Particle trajectories starting in the recharge zone at the top of the domain. The particle tracks are based on the conforming velocity approximation of the $\mathcal{RTN}_0$-projection for variably saturated flow on triangular prisms.

Figure 14 shows particle trajectories for the model of a rectangular recharge zone. The underlying velocity field in $\mathcal{RTN}_0$-space is mass-conservative and conforming. It is obtained by an extension of the $\mathcal{RTN}_0$-projection to variably saturated flow on triangular prisms in local coordinates. At the centroid of every recharge face at the top of the domain a particle is introduced and tracked through the domain. The particle tracks show a reasonable pattern: all particles reach the two possible outlets at the Dirichlet boundaries, and the expected symmetries of the trajectory patterns are overall achieved, although slight deviations from the expected pathlines can occasionally be observed.

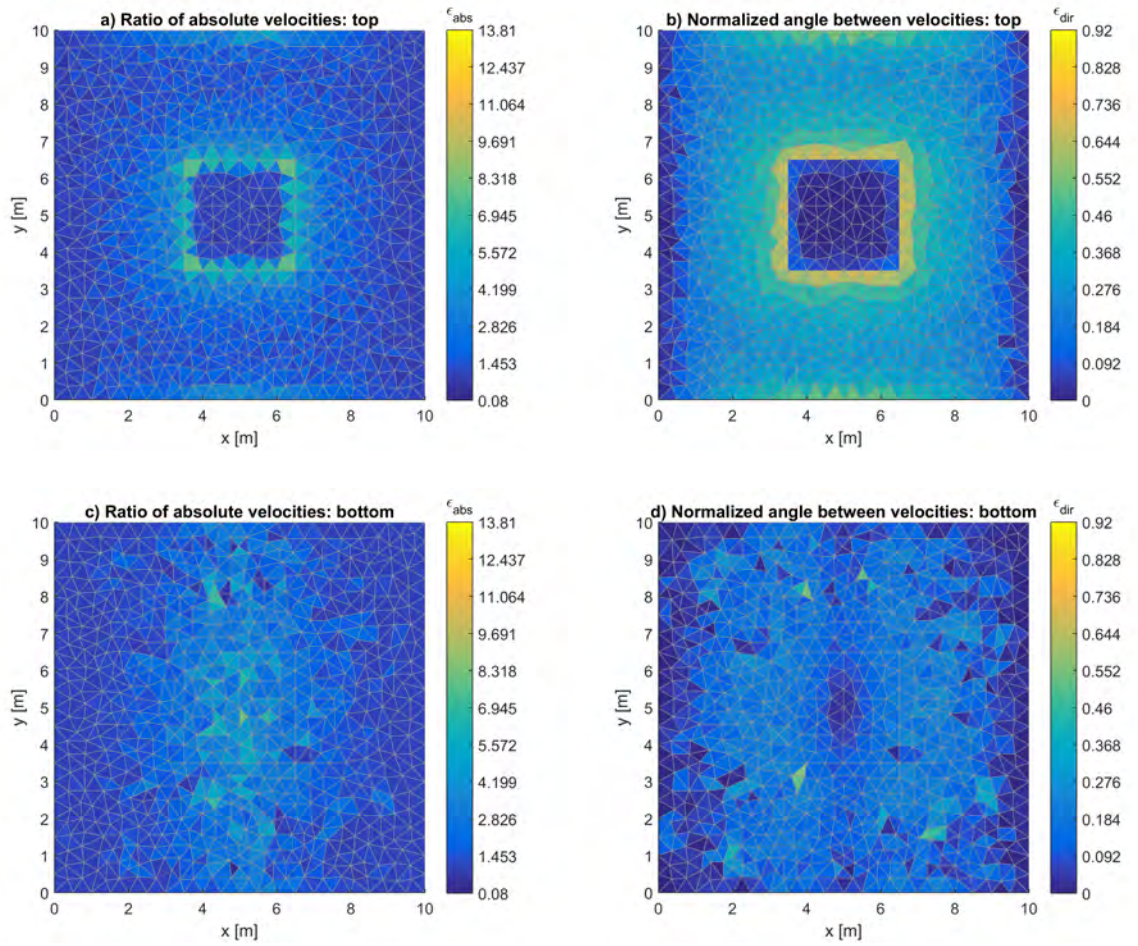

Figure 15: Difference measures between the primal finite-element-type velocity solution and an $\mathcal{RTN}_0$-projection for variably saturated flow on triangular prisms evaluated at the centroids of the top and the bottom element layers. a) Ratio of absolute velocities $\boldsymbol{\epsilon}_{abs}$ between the $\mathcal{RTN}_0$-projected velocity field over the finite-element-type primal velocity approximation at the centroids of the top element layer according to Equation 118; b) normalized angle between the velocities $\boldsymbol{\epsilon}_{dir}$ between the $\mathcal{RTN}_0$-projected velocity field and a finite-element-type primal velocity approximation at the centroids of the top element layer according to Equation 119; c) ratio of absolute velocities $\boldsymbol{\epsilon}_{abs}$ between the $\mathcal{RTN}_0$-projected velocity field over the finite-element-type primal velocity approximation at the centroids of the bottom element layer according to Equation 118; d) normalized angle between the velocities $\boldsymbol{\epsilon}_{dir}$ between the $\mathcal{RTN}_0$-projected velocity field and a finite-element-type primal velocity approximation at the centroids of the bottom element layer according to Equation 119.

Figure 15 shows the difference measures between the velocity field of the $\mathcal{RTN}_0$-projection and the finite-element-type primal velocity approximation for variably saturated flow at the centroids of triangular prisms. For the elements at the top depicted in the Figures 15a-b flow is unsaturated, while flow is saturated for Figures 15c-d. Considering the ratio of absolute velocities at the centroids depicted in Figure 15a it gets obvious that the magnitude of the $\mathcal{RTN}_0$-projected velocities within the recharge zone are very similar to the velocities of the primal solution. Only directly next to the recharge zone slightly larger differences can be observed, while the magnitudes further outside of the recharge zone are comparable to the primal approximation

again. However, it can be expected that differences between the primal solution and the $\mathcal{RTN}_0$-projection may be bigger next to the recharge zone because of larger differences in and different treatments of relative permeability, which is computed on the nodes in the primal solution procedure, and interpolated to the centroids of the elements in the $\mathcal{RTN}_0$-projection. Figure 15b depicts the difference in direction at the top of the domain. Here, it is obvious that the direction of the velocities within the recharge zone is nearly identical for the primal solution and the $\mathcal{RTN}_0$-projection. Larger differences occur especially directly next to the recharge zone, but also further outside of the recharge zone. However, it should be noted that also the finite-element-type primal velocity solution directly next to the recharge zone will be erroneous, and that, in general, there is only very little flow in the top elements outside of the recharge zone. Also, next to the no-flow boundaries at the top and the bottom of the domain the differences increase, where the flow is very little. Figure 15c depicts the ratio of the absolute velocities for the centroids of the bottom layer, where flow is saturated. Here, it can be seen that some velocities next to the flow divide at $x = 5\,\mathrm{m}$ are projected sometimes onto slightly higher absolute values compared to the primal velocity approximation, sometimes adjacent to elements, where the absolute values of the velocity are a little bit too low. The further away the centroids of the prisms are from the flow divide, the more the projected absolute values equal the absolute values of the primal velocity approximation. In Figure 15d the normalized angles between the projected velocity vector and the primal approximation of the velocity vector at the centroids of the bottom triangular prisms are depicted. It can be seen that the differences in direction are all small for saturated flow in the bottom elements, while differences in the center of the domain and towards the lateral Dirichlet boundary conditions are minimal.

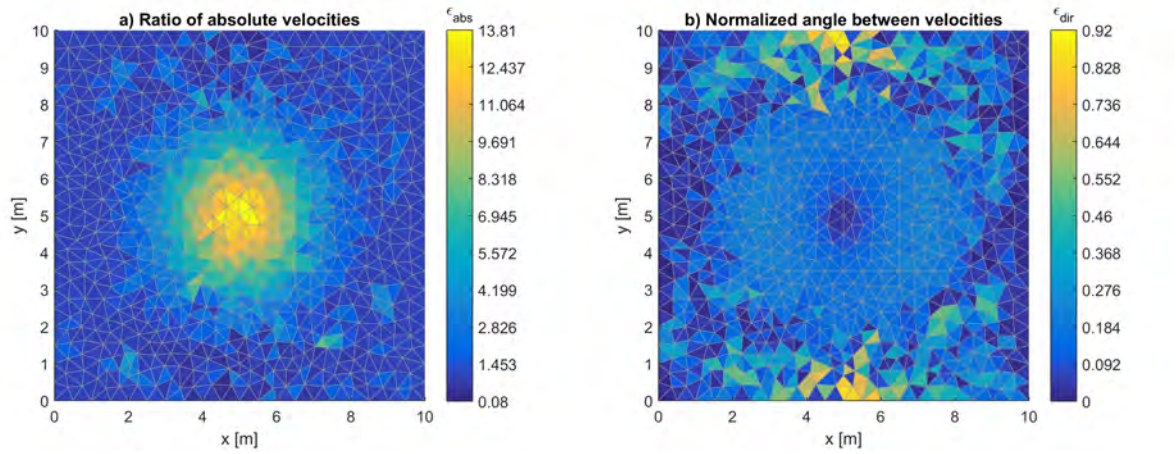

Figure 16: Difference measures between the primal finite-element-type velocity solution and an $\mathcal{RTN}_0$-projection for variably saturated flow on triangular prisms evaluated at the centroids of the element layer directly beneath the saturated-unsaturated interface. a) Ratio of absolute velocities $\boldsymbol{\epsilon}_{abs}$ between the $\mathcal{RTN}_0$-projected velocity field over the a finite-element-type primal velocity approximation at the centroids of the element layer directly below the saturated-unsaturated interface according to Equation 118; b) normalized angle between the velocities $\boldsymbol{\epsilon}_{dir}$ between the $\mathcal{RTN}_0$-projected velocity field and a finite-element-type primal velocity approximation at the centroids of the element layer directly below the saturated-unsaturated interface according to Equation 119.

In Figure 16a the difference measure of the ratio of the absolute velocities of the $\mathcal{RTN}_0$-projected velocity field over the absolute velocities of the primal solution is shown for the element layer directly below the saturated-unsaturated interface. Interestingly, the difference between both solutions is maximal in this element layer within the recharge square. Here, the magnitudes of the velocities of the $\mathcal{RTN}_0$-projection are about ten times larger than the ones of the primal solution, while the magnitudes outside the recharge square are again comparable for both solutions. Considering also the difference of the direction being the normalized angle between the $\mathcal{RTN}_0$-projected velocity and the primal approximation depicted in Figure 16b it gets obvious that the velocities of the $\mathcal{RTN}_0$-projection within the recharge zone still show a very similar direction compared to the velocity vectors of the primal solution following the physical direction of flow. Towards the top and bottom no-flow boundaries next to the flow divide at $x = 5\,\mathrm{m}$ some velocity vectors seem to be reverted. However, these elements exhibit nearly no flow, and the velocities are of very low magnitude within a subregion of homogenous material properties, and, therefore, of minor importance in the global optimization scheme. Moreover, also the primal solution is of minor accuracy in this region.

Table 8: Measures for the differences between finite-element-type primal velocity approximation and the extension of the $\mathcal{RTN}_0$-projection of the velocities for variably saturated flow on triangular prisms for the benchmark case of recharge via a square. As $\boldsymbol{\epsilon}_{abs}$ is a ratio, we give the respective infinity norm as $||(|1 - \boldsymbol{\epsilon}_{abs}|)||_\infty$.

	$mean(\cdot)$	$median(\cdot)$	$\|\|(\|\cdot\|)\|\|_\infty$
$\boldsymbol{\epsilon}_{abs}$	1.31	1.03	12.81
$\boldsymbol{\epsilon}_{dir}$	$2.64 \cdot 10^{-1}$	$2.05 \cdot 10^{-1}$	$9.2 \cdot 10^{-1}$

Table 8 gives an overview of the differences between the finite-element-type primal velocity approximation and the velocities of the extension of the $\mathcal{RTN}_0$-projection for variably saturated flow on triangular prisms for the benchmark case of recharge via a square. The mean of $\boldsymbol{\epsilon}_{abs}$ reflects the outliers also present in the associated infinity norm. Mean and median values of both difference measures may be still considered acceptable. However, the infinity norms show that there are also considerable differences in the solutions. Overall the differences between the solutions depicted in Table 8 are larger than the ones for saturated conditions including the cases for mild anisotropy (compare Tables 1, 2, 3, and 6). However, it is important to note that the primal velocity approximation is non-conforming, not mass conservative in the elements, and not even defined at the centroids by the primal solution procedure for variably saturated flow. That is, a deviation of the projected velocities from the primal solution is not only expected but desired.

3.2 Results for the Finite-Volume Flux Reconstruction

In this section, we depict and discuss the results of the FVM flux reconstruction and its associated particle trajectories. For steady-state saturated flow the FVM flux reconstruction would equal the cell-centered finite volume method already discussed (see section 2.4.3), if the weighting

procedures of hydraulic conductivity were equal, too. Because of that, we focus in this section on the finite-volume flux reconstruction as a postprocessing technique applied to variably saturated flow.

We evaluate the performance of our FVM flux reconstruction in two ways. First, we compare the hydraulic heads of the FVM reconstruction to the hydraulic heads of the primal solution obtained with HydroGeoSphere in the finite-difference mode (Panday et al., 1993; Therrien and Sudicky, 1996)(compare section 2.4.2). Note that the hydraulic heads of the FVM reconstruction are not used for any evaluation other than computing mass-conservative fluxes in $\mathcal{RTN}_0$-space suitable for accurate and consistent particle tracking. In the comparison of heads, we contrast the FVM hydraulic heads at the centroids of the elements with the HydroGeoSphere hydraulic heads at the centroids of the elements, evaluated by bilinear interpolation of the respective nodal values within the triangular prism.

The second demonstration of the performance of the FVM flux reconstruction is by illustrating consistent particle tracks resulting from this flux reconstruction compared to particle tracks based on the velocity field directly derived from the primal solution, such trajectories may stagnate because the underlying velocity field is non-conforming. We also compare the velocity estimate of the finite-element-type solution with the conforming, mass-conservative velocity approximation of the FVM flux reconstruction at the centroids of the elements. As the same benchmark model as in section 3.1.5 is used, this facilitates a direct comparison of the performance of the FVM flux reconstruction with the $\mathcal{RTN}_0$-projection for variably saturated flow. Furthermore, we apply our framework to a three-dimensional, variably saturated subsurface flow model connecting two floodplains close to Tübingen, Germany (Allgeier et al., 2020). By this we demonstrate the applicability of our framework to catchment-scale applications. The content and text of this section has been published to a large extent in Selzer et al. (2021).

3.2.1 Benchmark Model: Recharge in a Subarea of the Top Boundary

Our first benchmark model is the same as the model already considered in the section 3.1.5. Again, we consider a cubic domain having dimensions of $10\,\mathrm{m} \times 10\,\mathrm{m} \times 10\,\mathrm{m}$. The domain is discretized by 12,692 nodes and 22,572 elements in 18 elemental layers, the vertical discretization is refined at the top of the domain and around the height of the Dirichlet boundary conditions. On the left and right sides of the domain we assign homogeneous Dirichlet boundary conditions with a constant hydraulic head $h_{fix} = 6\,\mathrm{m}$. The front, back, and bottom sides are no-flow boundaries. A recharge zone is located at the center of the top of the domain. The zone is a square with dimensions of $3\,\mathrm{m} \times 3\,\mathrm{m}$, and the applied recharge rate is $200\,\mathrm{mm/a}$. We assume a uniform saturated hydraulic conductivity of $10^{-5}\,\mathrm{m/s}$, saturated and residual water contents of 0.4 and 0.065, respectively, and van Genuchten parameters $\alpha = 7.5\,1/\mathrm{m}$ and $N = 1.89$.

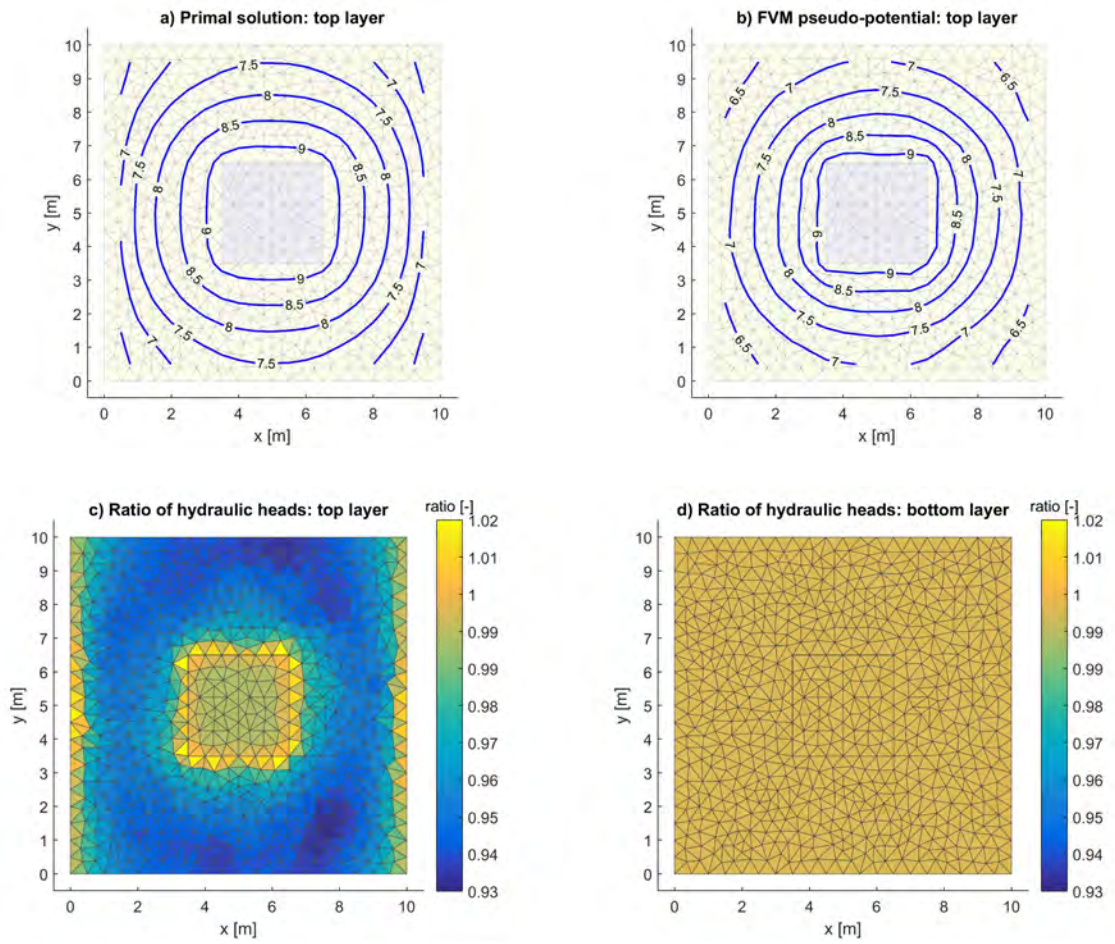

Figure 17: Benchmark problem with recharge via a square. Hydraulic heads for the primal solution linearly interpolated to the element centroid, h_c^{pri}, and according to the FVM flux reconstruction, h^{FVM}. a) Hydraulic heads according to the primal solution at the height of the top-layer element centroids; b) corresponding hydraulic heads of the FVM reconstruction; c) ratio h^{FVM}/h_c^{pri} of the former two hydraulic head solutions obtained at the height of the top-layer element centroids; d) ratio h^{FVM}/h_c^{pri} obtained at the height of the bottom-layer element centroids. (Taken from Selzer et al. (2021).)

Figure 17 illustrates simulated hydraulic heads according to the primal solution of HydroGeoSphere in finite difference mode (Figure 17a) and the FVM reconstruction (Figure 17b) for steady-state flow. The two hydraulic head solutions are similar, as quantified by the ratios in the top (Figure 17c) and bottom (Figure 17d) layers. It can be observed that the top-layer heads of the FVM reconstruction (Figure 17b) drop slightly faster towards the lateral boundaries of the domain than the heads of the primal solution (Figure 17a), leading to slightly lower hydraulic heads in the top layer for elements that are outside of the recharge zone. In the recharge zone and its direct vicinity, the head solutions are very close to each other. On the Dirichlet boundaries, the heads of both solutions are forced to be identical, leading to excellent agreement in the direct vicinity of those boundaries.

Within the saturated zone, illustrated by the solutions of the bottom layer (Figure 17d), the hydraulic heads computed by the two methods are identical within computational precision.

The excellent agreement may be attributed to the simple setup of the model. We would expect small differences between the primal solution and the FVM reconstruction because the solutions are based on different numerical methods. For instance, the primal solution is computed on the nodes of the elements, whereas the FVM reconstruction yields an average solution for each element.

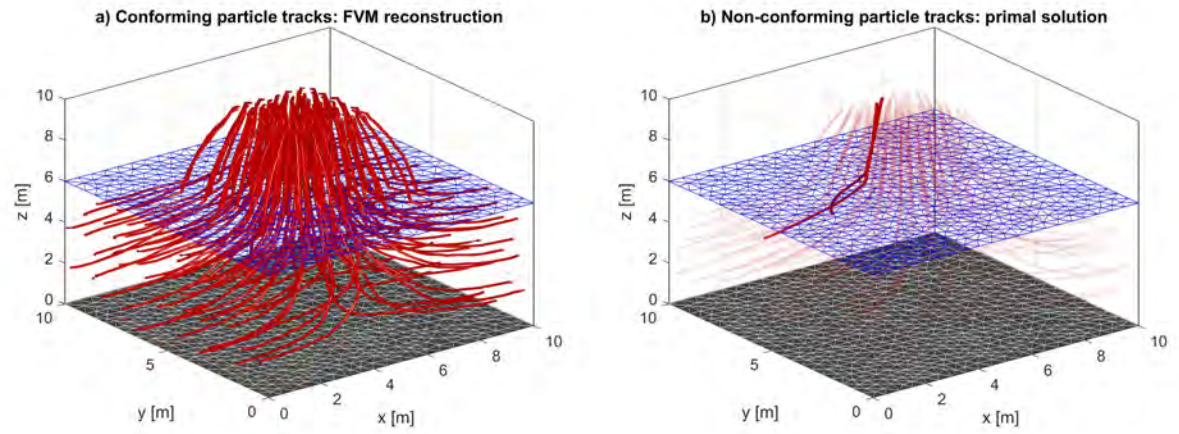

Figure 18: Benchmark problem with recharge via a square. Particle trajectories starting in the recharge zone at the top of the domain. a) particle tracks based on the conforming velocity approximation of the FVM flux reconstruction; b) trajectories based on the velocity of the finite-element-type primal solution, evaluated at the element centroids and assumed to be constant within the elements (transparent lines: regular particle tracks; opaque lines: two stagnating particle tracks). (Taken from Selzer et al. (2021).)

Figure 18 shows particle trajectories for the model of a rectangular recharge zone. At the centroid of every recharge face at the top of the domain a particle is introduced and tracked through the domain. Figure 18a shows the particle trajectories for the conforming velocity solution based on the FVM flux reconstruction. The particle tracks show a reasonable pattern: all particles reach the two possible outlets at the Dirichlet boundaries, and the expected symmetries of the trajectory patterns are overall achieved. Figure 18b shows the particle trajectories using a velocity field based on the primal FEM-type solution. For this plot, the velocity is evaluated at the centroid of each element and assumed constant within the element. While the patterns of the trajectories obtained by the two methods are overall similar, two particles stagnate in the FEM-type solution due to the non-conformity of the velocity field, which can lead to normal velocities at element faces that point outwards of each element. This prevents further tracking in a consistent manner.

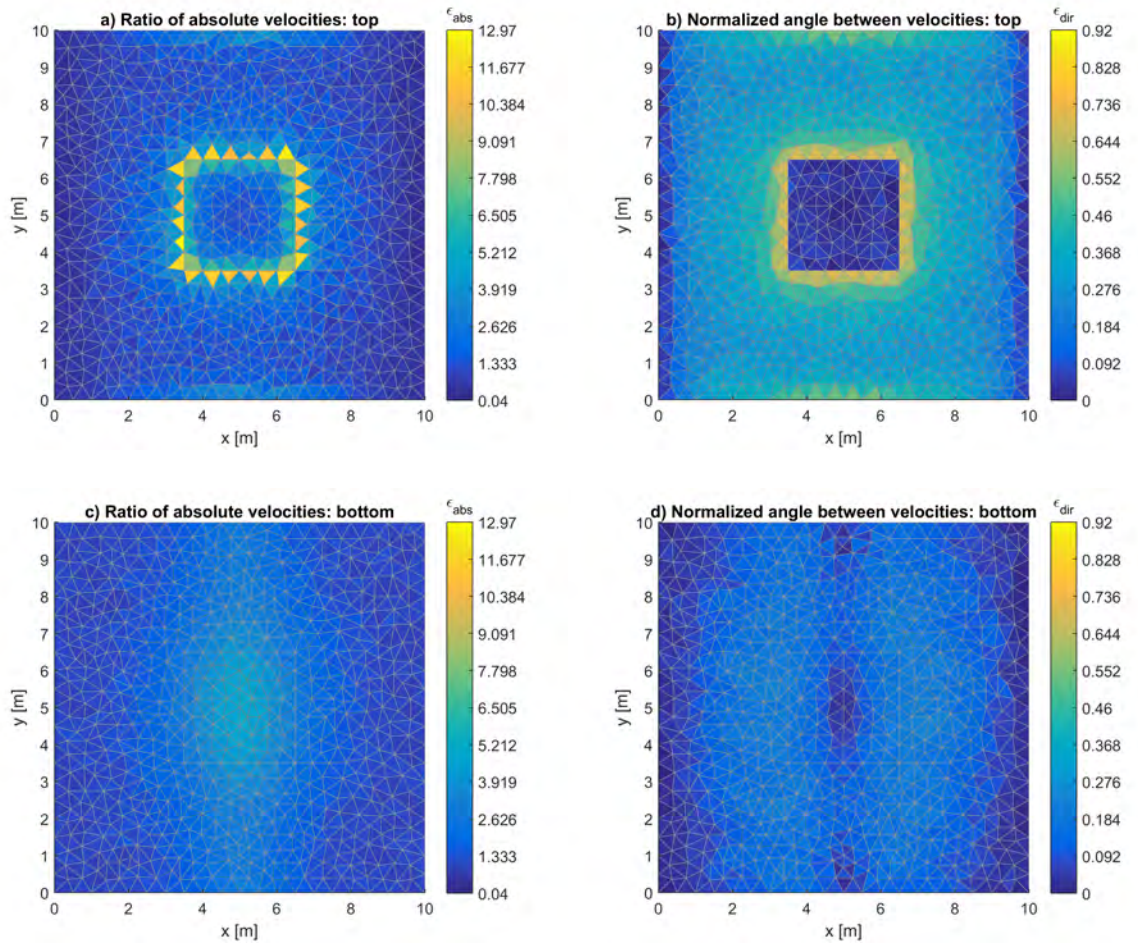

Figure 19: Difference measures between the primal finite-element-type velocity solution and an FVM flux reconstruction for variably saturated flow on triangular prisms evaluated at the centroids of the top and the bottom element layers. a) Ratio of absolute velocities $\boldsymbol{\epsilon}_{abs}$ between the velocity field of the FVM flux reconstruction over the finite-element-type primal velocity approximation at the centroids of the top element layer according to Equation 118; b) normalized angle $\boldsymbol{\epsilon}_{dir}$ between the velocity field of the FVM flux reconstruction and a finite-element-type primal velocity approximation at the centroids of the top element layer according to Equation 119; c) ratio of absolute velocities $\boldsymbol{\epsilon}_{abs}$ between the velocity field of the FVM flux reconstruction over the finite-element-type primal velocity approximation at the centroids of the bottom element layer according to Equation 118; d) normalized angle between the velocities $\boldsymbol{\epsilon}_{dir}$ between the velocity field of the FVM flux reconstruction and a finite-element-type primal velocity approximation at the centroids of the bottom element layer according to Equation 119.

Figure 19 shows the difference measures between the velocity field derived from the FVM flux reconstruction and a finite-element-type primal velocity approximation for variably saturated flow at the centroids of triangular prisms. For the elements at the top depicted in the Figures 19a-b flow is unsaturated, while flow is saturated for Figures 19c-d. Considering the ratio of absolute velocities at the centroids depicted in Figure 19a it gets obvious that the magnitude of the velocities of the FVM flux reconstruction within the recharge zone are overall very similar to the velocities of the primal solution. Only directly next to the border of the recharge zone larger differences can be observed, while the magnitudes further outside of the recharge zone are

again comparable to the primal approximation. However, it can be expected that differences between the primal solution and the velocities of the FVM flux reconstruction may be bigger next to the recharge zone because of larger differences in and different treatments of relative permeability, which is computed on the nodes in the primal solution procedure, and interpolated to the centroids of the elements in the FVM flux reconstruction including upstream weighting. Figure 19b depicts the difference in direction. Here, it is obvious that the direction of the velocities within the recharge zone is nearly identical for the primal solution and the velocities of the FVM flux reconstruction. Larger differences occur especially directly next to the recharge zone, but also further outside of the recharge zone. However, it should be noted that also the finite-element-type primal velocity solution directly next to the recharge zone will be erroneous, and that, in general, there is only very little flow in the top elements outside of the recharge zone. Also, next to the no-flow boundaries at the top and the bottom of the domain the differences increase slightly, where the flow is very small in particular. Figure 19c depicts the ratio of the absolute velocities for the centroids of the bottom layer, where flow is saturated. Here, it can be seen that some velocities next to the flow divide at $x = 5\,\mathrm{m}$ show slightly higher absolute values than the primal solution. The further away the centroids of the prisms are from the flow divide, the more the reconstructed absolute values equal the absolute values of the primal velocity approximation. In Figure 19d the normalized angles between the reconstructed velocity vector and the primal approximation of the velocity vector at the centroids are depicted. It can be seen that the differences in direction are all small for saturated flow in the bottom elements, while some smaller differences occur showing a symmetric pattern around the flow divide at $x = 5\,\mathrm{m}$. Overall the pattern depicted in Figure 19 is very similar to those of the $\mathcal{RTN}_0$-projection shown in Figure 15.

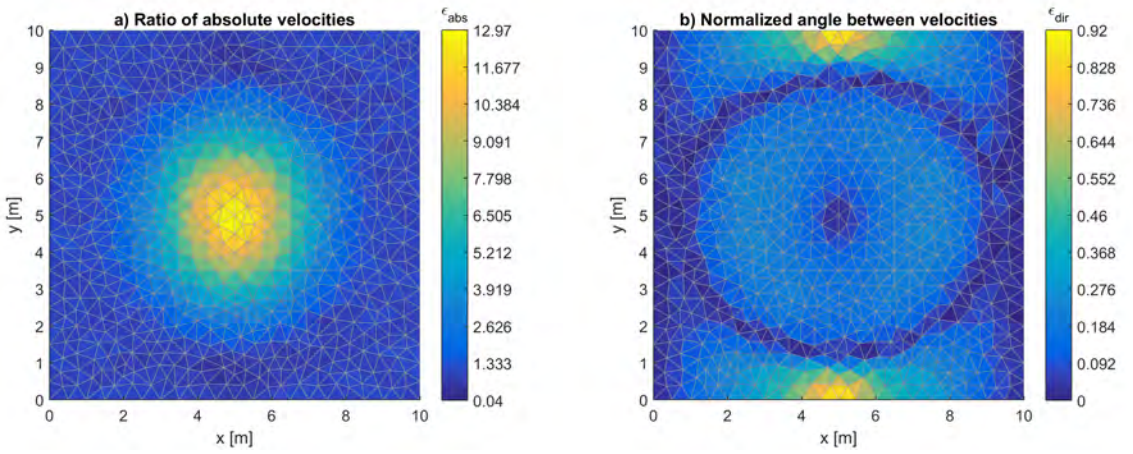

Figure 20: Difference measures between the primal finite-element-type velocity solution and the velocity field based on the FVM flux reconstruction for variably saturated flow on triangular prisms evaluated at the centroids of the element layer directly beneath the saturated-unsaturated interface. a) Ratio of absolute velocities $\boldsymbol{\epsilon}_{abs}$ between the velocity field based on the FVM flux reconstruction over the finite-element-type primal velocity approximation at the centroids of the element layer directly below the saturated-unsaturated interface according to Equation 118; b) normalized angle between the velocities $\boldsymbol{\epsilon}_{dir}$ between the velocity field of the FVM flux reconstruction and a finite-element-type primal velocity approximation at the centroids of the element layer directly below the saturated-unsaturated interface according to Equation 119.

In Figure 20a the difference measure of the ratio of the magnitudes of the velocities of the FVM flux reconstruction over the absolute velocities of the primal solution is shown for the element layer directly below the saturated-unsaturated interface. Interestingly, the difference between both solutions is maximal again in this element layer within the recharge square like for the comparison of the $\mathcal{RTN}_0$-projection with the finite-element-type primal solution (see Figure 16). The magnitudes of the velocities of the FVM flux reconstruction are about ten times larger, while the magnitudes outside the recharge square are again comparable for both solutions, which is similar to the comparison of the $\mathcal{RTN}_0$-projection with the finite-element-type primal solution. Considering also the difference of the direction being the normalized angle between the velocity fields of the FVM flux reconstruction and the primal approximation depicted in Figure 20b it gets obvious that the velocities of the FVM flux reconstruction within the recharge zone still show a very similar direction compared to the velocity vectors of the primal solution following the physical direction of flow. Outside the recharge square differences slightly increase. Moreover, there is a circular pattern around the recharge zone with differences in direction being close to zero. Towards the top and bottom no-flow boundaries next to the flow divide at $x = 5\,\mathrm{m}$ some velocity vectors seem to be reverted compared to the primal velocity approximation. This is again similar to the findings of the $\mathcal{RTN}_0$-projection for the same benchmark case depicted in Figure 16.

Table 9: Measures for the difference between the finite-element-type primal velocity approximation and the extension of the FVM reconstruction for variably saturated flow on triangular prisms for the benchmark case of recharge via a square. As $\boldsymbol{\epsilon}_{abs}$ is a ratio, we give the respective infinity norm as $||(|1 - \boldsymbol{\epsilon}_{abs}|)||_\infty$.

	$mean(\cdot)$	$median(\cdot)$	$\|\|(\|\cdot\|)\|\|_\infty$
$\boldsymbol{\epsilon}_{abs}$	1.17	0.93	11.97
$\boldsymbol{\epsilon}_{dir}$	$2.60 \cdot 10^{-1}$	$1.75 \cdot 10^{-1}$	$9.17 \cdot 10^{-1}$

Table 9 gives an overview of the difference measures between the finite-element-type primal velocity approximation and the velocities based on the FVM flux reconstruction for variably saturated flow on triangular prisms for the benchmark case of recharge via a square. The mean of $\boldsymbol{\epsilon}_{abs}$ reflects the outliers also present in the associated infinity norm. Mean and median values of both difference measures may be still considered acceptable. However, the infinity norms reflect that there are also considerable differences in the solutions. Overall, the differences between the solutions depicted in Table 9 are similar to those of the $\mathcal{RTN}_0$-projection depicted in Table 8 although slightly smaller. However, it is important to note that the primal velocity approximation is non-conforming and not mass conservative. That is, a deviation of the postprocessed velocities from the primal solution is not only expected but again desired.

Overall, it is remarkable how similar the solutions of the $\mathcal{RTN}_0$-projection and the FVM flux reconstruction are. However, we also attribute this to the very simple setup of the benchmark case and would expect larger differences for more complicated scenarios.

3.2.2 Case Study: Particle Trajectories in a Floodplain

We now apply our FVM flux reconstruction and particle-tracking scheme to a subsurface flow model on the scale of a sub-catchment. The model was originally developed for a stochastic analysis of groundwater divides (Allgeier et al., 2020). In the following, we use a single realization of the original ensemble to demonstrate the applicability of our scheme.

The model simulates variably saturated flow in the Ammer floodplain and its surroundings close to Tübingen in South-Western Germany (Martin et al., 2020). The domain is discretized by 74,412 nodes and 138,565 deformed triangular prisms (35 layers of 3959 elements each). Like in the benchmark case, we use HydroGeoSphere in finite-difference mode to obtain the steady-state solution of Richards' Equation 6. Towards this end, we simulate transient flow with constant forcings until steady state was reached. Based on this solution, we perform our FVM flux reconstruction and the particle tracking.

The subsurface is subdivided into different hydrostratigraphic units, following the layered structure of an underlying regional geological model (D'Affonseca et al., 2020) and studies within the floodplain (Martin et al., 2020). The bedrock consists of a sequence of mud- and sandstones, with a weathering layer of higher hydraulic conductivity. Quaternary fillings in the valleys form alluvial aquifers separated by clay layers. We consider the differences of hydraulic conductivity between the hydrostratigraphic units to be higher than the internal variability and assigned each

unit a uniform set of hydraulic parameters. Saturated hydraulic conductivities range between $\approx 5 \times 10^{-10}$ m/s (vertical conductivity in the top colluvial clay) and $\approx 3 \times 10^{-4}$ m/s (horizontal conductivity in the gravel aquifer of the floodplain). Vertical hydraulic conductivities of the units differ from their horizontal counterparts. For the sake of simplicity we approximate the saturated hydraulic conductivities of the mobility coefficients (Equation 65) by taking the vertical hydraulic conductivities for the vertical connections, and the horizontal hydraulic conductivities for the horizontal connections.

We assign Dirichlet boundary conditions at the inlet and outlet faces of the floodplain filling, as well as at the southern boundary of the domain. At the top of the domain, we assign Neumann boundary conditions according to the net infiltration rates, which varies between 20 and 60 mm/a depending on land use. Robin boundary conditions are used to simulate the exchange between groundwater and the Ammer River, as well as the seepage of groundwater into artificial drainage channels. All other boundaries are no-flow boundaries. All boundary conditions are assumed to be constant over time, and we simulate subsurface flow until steady state is reached.

Figure 21 presents contourlines of hydraulic heads according to the primal solution and the FVM reconstruction at the element centroids at the top and bottom of the domain. The two hydraulic head fields are very similar. At the top of the domain (Figures 21a and 21b), the system is mostly unsaturated. In particular, towards the west the depth to the water table is fairly large. Here the values of hydraulic head differ the most between the primal solution and the FVM reconstruction. At the bottom of the domain (Figures 21c and 21d), the system is fully water-saturated. Also here we see some differences in the hydraulic-head distribution. However, the major objective of the FVM reconstruction is to obtain a conforming velocity field rather than exactly reproducing the hydraulic head field of the finite-element-type simulation by finite volumes.

We use the FVM reconstruction to delineate various sub-catchments by particle tracking. Towards this end, we place a particle at the centroid of each top face of the domain and track the particle until it has reached an outflow boundary. In total these are 3959 particle tracks visualized in Figure 22.

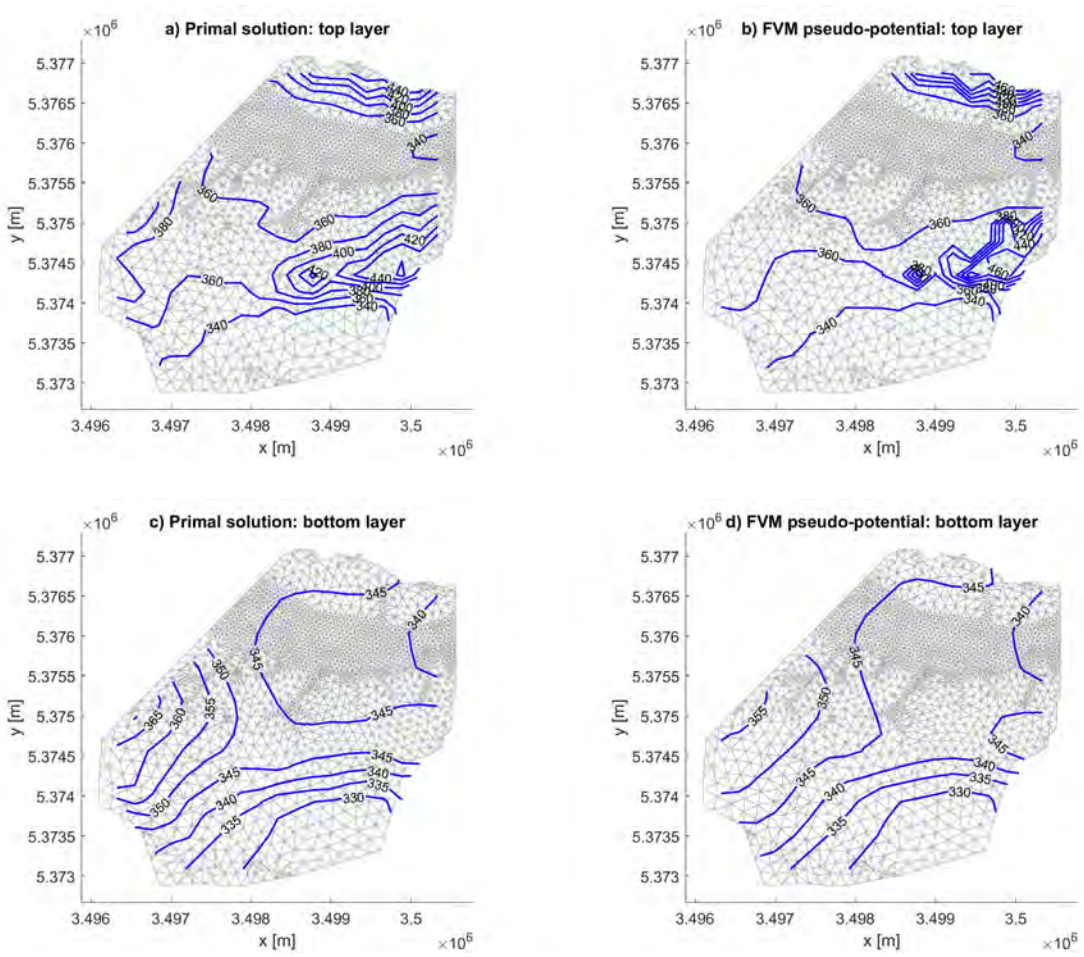

Figure 21: Application to regional subsurface flow: comparison between hydraulic heads of the primal solution (a & c) and the FVM reconstruction (b & d). a) & b): hydraulic heads in the top layer; c) & d): hydraulic heads at the bottom of the domain. Coordinates are given in the Gauss-Krüger coordinate system; hydraulic heads are given in meters above sea level. (Taken from Selzer et al. (2021).)

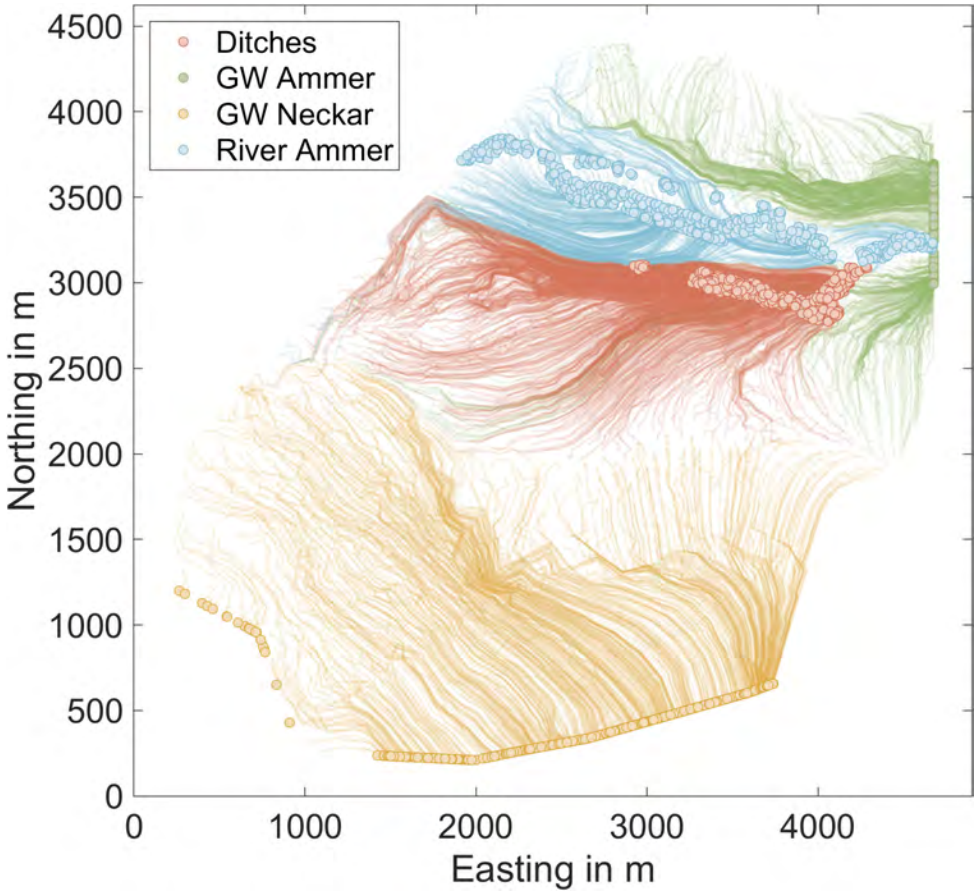

Figure 22: Trajectories of particles released at the top of the domain. Circles: ending locations. Colors reflect the type of outflow boundary reached: drainage channels (dusky pink), Ammer River (light blue), groundwater outflow within the Ammer valley (green), groundwater outflow in the Neckar valley (orange). (Taken from Selzer et al. (2021).)

Figure 22 shows a two-dimensional horizontal projection of the resulting particle tracks. The trajectories are colored according to the outlet where they end up. A local groundwater divide becomes apparent between all orange particle tracks reaching the southern groundwater outlet in the Neckar valley, and the remaining particles ending up in the Ammer valley. All inserted particles reach an outflow boundary, that is, no particles get stuck in the domain or exit at non-physical destinations.

In practice, the Tecplot™ (Tecplot Inc., 2019) visualization and analysis platform is often used as visualization software for HydroGeoSphere results. Tecplot™ can trace streamlines using the element-wise given velocity field. Although this visualization software is not designed for rigorous and consistent particle tracking, a qualitative comparison of the streamlines generated by Tecplot™ revealed a good general agreement regarding the overall pattern with our particle tracking approach. However, like in our first benchmark case, in which we used a velocity field derived from the primal solution at the centroids (Figure 18), some particles released at the surface do not reach an outlet boundary condition when tracked by Tecplot™. Instead they stagnate within the domain, or erroneously reach a no-flow boundary, or even an inflow boundary. Moreover, we deem our trajectories to be more accurate. The major advantage of consistent

particle tracking based on conforming velocity fields is physical and numerical accuracy of the flow paths, which avoids non-physical exit locations or stagnation of particles.

Table 10: Wall and CPU times of the different computational steps. All computations are executed on a Dell Inspiron 7570 personal computer operating under Ubuntu 20.04 LTS (Processor: Intel i7-8550U (8) @ 4.000GHz, Memory: 16 GB). Numbers are displayed as "average ± standard deviation", and derived from a sample of size $n = 10$. (Taken from Selzer et al. (2021).)

Computation	Wall time in s	CPU time in s
HydroGeoSphere	543.6 ± 4.6	1085.2 ± 9.1
Geometrical computations	16.8 ± 0.2	31.6 ± 0.3
FVM flux reconstruction	25.6 ± 0.4	32.7 ± 0.4
Particle tracking	10.7 ± 0.1	17.3 ± 0.2

Table 10 summarizes the wall-clock and CPU times needed for achieving the primal solution as well as the times needed for the finite volume flux reconstruction. Here we differentiate between the effort to perform geometry-related computations and the actual FVM flux reconstruction. For an increased representativeness we report the averages and the standard deviations of run times for ten model runs each. Additionally, we give the time needed to compute the trajectories of all 3959 individual particles on the CPU.

The results of Table 10 clearly show that computing the primal solution is the time limiting step in the computations. The postprocessing as well as the particle tracking take only about a tenth of the time which is needed for computing the primal solution.

4 Conclusions and Outlook

4.1 Conclusions

Accurate and consistent particle tracking relies on conforming velocity fields, which locally preserve mass in the elements and yield a continuous normal component of the Darcy velocity on the element faces (Selzer et al., 2021). As published in Selzer and Cirpka (2020), I have proposed an $\mathcal{RTN}_0$-projection of fluxes obtained from solving the groundwater flow equation by $\mathcal{P}_1$ Galerkin FEM. The projection yields physically reasonable flow fields preserving the smoothness of the $\mathcal{P}_1$ Galerkin FEM solution for isotropic test cases and mild anisotropy on triangles and tetrahedra. For comparison, I have presented an analytical solution and a cell-centered finite-volume formulation on simplices in which I account for Dirichlet and Neumann boundary conditions by ghost nodes.

The results of the $\mathcal{RTN}_0$-projection are similar to those obtained by an analytical solution and the cell-centered finite volumes for two-, and three-dimensional isotropic and mildly anisotropic saturated flow fields. The velocity approximations of the $\mathcal{RTN}_0$-projection and of the FVM are in the same function space, and the velocity fields are similar in magnitude and the direction for saturated flow on simplices. By construction, the velocity fields of the $\mathcal{RTN}_0$-projection are conforming and mass-conservative. In comparison to velocity fields obtained from cell-centered FVM, the projected fields are smoother and trajectories are less angular so that I assess the projected velocity fields in general to be physically more accurate than the fields obtained by the FVM scheme for saturated flow, at least in regions of uniform hydraulic conductivity. However, the computational costs of a cell-centered FVM scheme are much lower than those of the $\mathcal{RTN}_0$-projection. For an FVM solution one system of equations on the order of elements has to be solved, in which the matrix of the basic discretization is positive definite and symmetric. By contrast, the $\mathcal{RTN}_0$-projection requires first solving a system of equations on the order of the number of nodes with symmetric, positive-definite matrix to obtain the $\mathcal{P}_1$ Galerkin FEM solution, followed by solving a saddle-point problem in the $\mathcal{RTN}_0$-projection step, in which the matrix is symmetric, non-definite, and on the order of the number of elements plus the number of faces without no-flow boundary conditions. Both, the projected and the lowest-order finite-volume velocity fields are in $\mathcal{RTN}_0$-space. For large-scale applications, the computational advantage of the FVM scheme may outweigh the smoothness of the $\mathcal{RTN}_0$-projection. Conversely, too variable velocity fields may cause erroneous twisting of streamlines in heterogeneous formations, which can have significant effects on the simulation of transverse mixing (Cirpka et al., 2015; Bennett et al., 2017).

I claim that the $\mathcal{RTN}_0$-projection, like the flux correction method of Odsæter et al. (2017), gives better results than the postprocessing methods of Schiavazzi (2013) and Scudeler et al. (2016), which are applications of the schemes by Larson and Niklasson (2004) and Kees et al. (2008), in which Darcy velocity differences between the $\mathcal{P}_1$ Galerkin solution and the $\mathcal{RTN}_0$-projection showed much larger relative differences in low-flow regions. The latter can be avoided by minimizing differences in hydraulic gradients rather than Darcy velocities between the $\mathcal{P}_1$

Galerkin FEM and the $\mathcal{RTN}_0$-solutions. While Odsæter et al. (2017) considered hydraulic gradients on element faces, I minimize the difference in hydraulic gradients at the element centroids. This approach yields physically reasonable velocity fields regarding the magnitude of the velocity vectors and does not introduce erroneous numerical rotation, neither for two-dimensional nor for three-dimensional saturated flow fields with isotropic or mildly anisotropic hydraulic conductivity, which seems to occur in other flux correction schemes even in the isotropic case (Schiavazzi, 2013), and have occasionally been observed in $\mathcal{RTN}_0$-based mixed (hybrid) FEM schemes (Hoteit et al., 2002).

Consistency and empirical convergence could be shown for isotropic and anisotropic cases comparing the $\mathcal{RTN}_0$-projected solution with an analytical solution of groundwater flow on a unit square, where water enters and exits over windows on the boundary of the square opposite to each other. However, if stronger anisotropy is considered, the $\mathcal{RTN}_0$-projection may numerically induce rotation and wrong magnitudes of the velocities leading to partial failure of the scheme in low-flow regions, although empirical convergence, yet with varying order, can be observed. Moreover, an extension and application of the $\mathcal{RTN}_0$-projection to a finite-element-type primal solution of variably saturated flow on triangular prisms in a local coordinate system was successfully achieved using contravariant Piola-transformation (Rognes et al., 2009).

Transient flow can be accounted for in a straight-forward manner by treating the change of storage like a source/sink term in steady-state flow. In order to do this, however, the nodal loads computed by the primal finite-element-type method must be distributed to the elements associated with the nodes. However, especially for non-linear flow laws like Richards' equation the application of a postprocessing scheme, like the $\mathcal{RTN}_0$-projection for mapping an existent non-conforming flow solution onto a conforming one, can still be computationally much cheaper than computing a conforming flow solution from scratch using an $\mathcal{RTN}_0$ MFEM or comparable methods. As a postprocessor, the $\mathcal{RTN}_0$-projection has to be performed only at time points at which a velocity field is wanted, and the non-linear dependence of hydraulic conductivity on the simulated head- and saturation-fields is already solved within the primary solution step, deeming the projection itself linear.

In addition to the $\mathcal{RTN}_0$-projection, I have also presented a postprocessing scheme for non-conforming velocity fields by reconstructing the hydraulic head and associated flux distribution by cell-centered finite volumes of lowest-order as published in Selzer et al. (2021). This flux reconstruction is stable, can be coupled to a finite-element-type primal solution of variably saturated flow, and does not induce severe numerical artifacts like reverted flow in sub-regions of the domain, or too large velocities by orders or magnitude. Moreover, the resulting system of equations is only on the order of the number of elements and positive definite, which makes solving the equation system of the FVM flux reconstruction much easier and quicker than the one of the $\mathcal{RTN}_0$-projection. Also, the FVM flux reconstruction does not require transformations of coordinates like the $\mathcal{RTN}_0$-projection on non-simplicial elements reducing the computational overhead significantly. This makes the FVM flux reconstruction more attractive for catchment-scale applications.

I showed that the FVM flux reconstruction yields hydraulic-head fields which are close to the primal solution computed by HydroGeoSphere in finite-difference mode (Therrien and Sudicky, 1996). Moreover, the FVM flux reconstruction yields very similar results compared to those of the $\mathcal{RTN}_0$-projection when applied to a benchmark case of variably saturated flow on triangular prisms. In the FVM reconstruction, I make use of the relative permeabilities of the primal solution, assuming that the latter already accounts for the non-linearity of Richards' equation. Also, I consider storage changes of the primal solution as known divergence in an otherwise steady-state simulation in the FVM reconstruction. This is computationally much faster than computing a conforming solution from scratch, or the $\mathcal{RTN}_0$-projection. Compared to preceding flux corrections (e.g., Larson and Niklasson, 2004; Scudeler et al., 2016; Selzer and Cirpka, 2020; Sun and Wheeler, 2006), the proposed scheme is absolutely stable and does not lead to severe numerical artifacts like velocities being too high by orders of magnitude in low-velocity regions, or reverted velocities in such regions (compare Schiavazzi, 2013; Selzer and Cirpka, 2020). These advantages come at the cost of accurately mimicking the primal model, including all material properties and boundary conditions, in the FVM reconstruction.

Based on a conforming velocity field in $\mathcal{RTN}_0$-space, I formulated semi-analytical particle tracking schemes relying on element-wise, analytical solutions for the particle trajectories in $\mathcal{RTN}_0$-space on triangles, tetrahedra, and triangular prisms. The particle tracking schemes on triangles and tetrahedra are formulated in global, physical coordinates, although their application in a local coordinate system is possible, too. For triangular prisms, I have derived an analytical solution in a mixed coordinate system, which is exact for non-deformed triangular prisms.

For the particle-tracking scheme in mixed coordinates, the inaccuracy increases the more deformed the triangular prism is and the more the bottom and the top faces are tilted towards each other, as well as the more the height of the element at the centroid differs from the mean height along the trajectory. However, I think that in catchment-scale hydrological modeling the major uncertainty is in the approximation of material properties, uncertain boundary conditions, and in simplifications of the physical representation such as neglecting preferential flow paths. In contrast, the prisms normally used in catchment-scale subsurface hydrological modeling are often only moderately deformed. Therefore, I believe that the additionally introduced inaccuracy is small compared to the other inaccuracies present in most catchment-scale variably saturated flow models, and the approximation above is justified by the gain in computational efficiency.

I have shown that the FVM flux reconstruction as well as the associated particle-tracking scheme on potentially deformed triangular prisms can be applied to realistic subsurface flow models on the catchment scale, including varying element geometries, several geological layers, and heterogeneous boundary conditions. The scheme is accurate, gives results close to the primal solution in the hydraulic head reconstruction, and avoids gaps in definition. The particle trajectories can be used for the delineation of capture zones or for streamline-based simulations of reactive transport, making use of the finite-element-type primal solution of catchment-scale variably saturated subsurface flow.

4.2 Outlook

Based on the outcome of this thesis, and the conclusions drawn, future improvements and fields of research may include the following.

- The FVM flux reconstruction was investigated in detail and has been successfully applied to variably saturated flow in porous media on the catchment scale. In future research, also the $\mathcal{RTN}_0$-projection could be extended such that it would be applicable to the catchment scale, maybe including a multi-point projection for non-simplicial elements. While the FVM flux reconstruction is already speed-optimized, also the $\mathcal{RTN}_0$-projection could be further speed optimized to make an application on the catchment scale more feasible. However, it remains an open question, if the $\mathcal{RTN}_0$-projection is stable enough to be successfully applied to catchment-scale applications. In any case the run times for the $\mathcal{RTN}_0$-projection will be longer, which could only be justified by an increase in accuracy.
- It has been shown that the FVM flux reconstruction is a suitable method for postprocessing non-conforming primal flux solutions as a base for accurate and consistent particle tracking. However, also an application of the postprocessed velocity fields for numerical solutions of the advection-dispersion equation by a cell-centered Eulerian method could be investigated. It can be expected that a cell-centered numerical solution of the advection-dispersion equation employing a postprocessed conforming velocity field will perform better, than a solution employing a non-conforming velocity field; although numerical dispersion will be a problem in a solution based on the velocities of a lowest-order FVM flux reconstruction.
- So far, the flux postprocessing schemes have only been developed for variably saturated flow in porous media. However, integrated hydrosystem models typically couple porous media flow with overland flow, river flow, flow in fractures, or flow in other subdomains. Overland flow, and river flow are sometimes approximated by the diffusive wave approximation, which has the shape of a non-linear advection-dispersion equation including a source/sink term. Otherwise flow may be conceptualized as a full free-flow problem employing the Navier-Stokes equations, or one may describe creeping flow in subdomains via the Stokes equation. For full integrated hydrosystem modelling one would need to define an FVM flux reconstruction or an other postprocessing method like the $\mathcal{RTN}_0$-projection for all flow regimes occurring and coupled to each other, if the according flow equations were solved with a primal numerical method leading to non-conforming fluxes violating element-wise mass conservation.
- If different hydrosystems are considered within a model, a coupling strategy needs to be defined for a flux postprocessing technique. Also, the particle tracking schemes would need to be able to handle the transition between different hydrosystems. This is in particular interesting, if hydrosystems are conceptualized in different dimensions. For instance, one may conceptualize variably saturated flow in porous media in three spatial dimensions, while overland flow may be described as a two-dimensional problem embedded in three dimensions following the terrain topography. Particles which leave a higher-dimensional

space like the porous medium to enter the overland flow domain are straight-forward to track. However, the reverse case is difficult as particles within a two-dimensional flow field can not leave this domain in a deterministic manner as normal to the dimensions considered there is no velocity component approximated. That is, there is no velocity pointing out of the overland flow domain for the flow solution within the overland flow domain. Particles could only leave a lower dimensional domain like the overland flow domain in a stochastic way based on the exchange fluxes between the domains.

- The FVM flux reconstruction is based on a lowest-order FVM and an adapted two-point flux approximation scheme including a scaling factor such that only the orthogonal part of a mobility coefficient normal to an internal face is considered. Anisotropy as a full tensor is not accounted for in the described scheme, and anisotropy as a diagonal tensor can only be considered in a fully consistent manner, if the grid is aligned with the principal directions of a diagonal hydraulic-conductivity tensor. A possible extension would be to account for anisotropy in the flux approximation. In order to do this, the present two-point flux approximation scheme could be, for instance, replaced by a multi-point flux approximation being the base for a better approximation of anisotropy (compare e.g., Su et al. (2020)), or a scheme similar to the ones of Edwards (2002) and Friis et al. (2009) could be employed using transformations of coordinates.
- Besides of using lowest-order finite volumes or the $\mathcal{RTN}_0$-projection for approximating a conforming and mass-conservative velocity field, one may also reconstruct a conforming velocity field using a discontinuous Galerkin method. Although, the resulting velocity field of a standard discontinuous Galerkin method is non-conforming for all orders higher than lowest order, the average flux over the element boundary is mass conservative in a discontinuous Galerkin method (Bastian and Rivière, 2003). An element-wise projection of the flux originated by a DG method is possible (Bastian and Rivière, 2003; Vidotto et al., 2018). The use of a discontinuous Galerkin method for flux reconstruction could reduce the effects of numerical dispersion present in a lowest-order FVM reconstruction and it could facilitate an elegant way to account for material anisotropy.
- The concept of the discontinuous Galerkin method could also be employed for a velocity projection. One could compute the element-wise defect in mass conservation, and distribute this residual in a discontinuous-Galerkin sense along the element boundaries. Based on this, an element-wise projection of the approximated normal flux on the element boundaries onto a conforming velocity field in $\mathcal{RTN}_0$-space, or a higher-order velocity space is possible (Bastian and Rivière, 2003; Vidotto et al., 2018).

Bibliography

Allgeier, J., González-Nicolás, A., Erdal, D., Nowak, W., and Cirpka, O. A. (2020). A Stochastic Framework to Optimize Monitoring Strategies for Delineating Groundwater Divides. *Frontiers in Earth Science*, 8(554845). https://doi.org/10.3389/feart.2020.554845.

Aquanty, Inc. (2015). HydroGeoSphere. A three-dimensional numerical model describing fully-integrated subsurface and surface flow and solute transport: User manual. Technical report, Aquanty, Inc., ON, Ontario.

Atchley, A. L., Maxwell, R. M., and Navarre-Sitchler, A. K. (2013). Using streamlines to simulate stochastic reactive transport in heterogeneous aquifers: Kinetic metal release and transport in CO2 impacted drinking water aquifers. *Advances in Water Resources*, 52:93 – 106. https://doi.org/10.1016/j.advwatres.2012.09.005.

Bahriawati, C. and Carstensen, C. (2005). Three Matlab Implementations of the Lowest-order Raviart-Thomas Mfem with a Posteriori Error Control. *Computational Methods in Applied Mathematics*, 5(4):333–361. https://doi.org/10.2478/cmam-2005-0016.

Bastian, P. and Rivière, B. (2003). Superconvergence and $H(\mathrm{div})$ projection for discontinuous Galerkin methods. *International Journal for Numerical Methods in Fluids*, 42(10):1043 – 1057. https://doi.org/10.1002/fld.562.

Bear, J. and Cheng, A. H.-D. (2010). *Modeling Groundwater Flow and Contaminant Transport*. Vol. 23 of Theory and Applications of Transport in Porous Media. Springer, Heidelberg. https://doi.org/10.1007/978-1-4020-6682-5.

Bellin, A., Salandin, P., and Rinaldo, A. (1992). Simulation of dispersion in heterogeneous porous formations: Statistics, first-order theories, convergence of computations. *Water Resources Research*, 28(9):2211–2227. https://doi.org/10.1029/92WR00578.

Bennett, J. P., Haslauer, C. P., and Cirpka, O. A. (2017). The impact of sedimentary anisotropy on solute mixing in stacked scour-pool structures. *Water Resour. Res.*, 53(4):2813–2832. https://doi.org/10.1002/2016WR019665.

Brezzi, F., Douglas, J., Durán, R., and Fortin, M. (1987). Mixed finite elements for second order elliptic problems in three variables. *Numerische Mathematik*, 51(2):237–250. https://doi.org/10.1007/BF01396752.

Brezzi, F., Douglas, J., and Marini, L. D. (1985). Two families of mixed finite elements for second order elliptic problems. *Numerische Mathematik*, 47(2):217–235. https://doi.org/10.1007/BF01389710.

Celia, M. A., Bouloutas, E. T., and Zarba, R. L. (1990). A general mass-conservative numerical solution for the unsaturated flow equation. *Water Resources Research*, 26(7):1483 – 1496. https://doi.org/10.1029/WR026i007p01483.

Cirpka, O. A., Chiogna, G., Rolle, M., and Bellin, A. (2015). Transverse mixing in three-

dimensional nonstationary anisotropic heterogeneous porous media. *Water Resources Research*, 51(1):241 – 260. https://doi.org/10.1002/2014WR015331.

Cirpka, O. A., Frind, E. O., and Helmig, R. (1999). Numerical methods for reactive transport on rectangular and streamline-oriented grids. *Advances in Water Resources*, 22(7):711 – 728. https://doi.org/10.1016/S0309-1708(98)00051-7.

Cordes, C. and Kinzelbach, W. (1992). Continuous groundwater velocity fields and path lines in linear, bilinear, and trilinear finite elements. *Water Resources Research*, 28(11):2903 – 2911. https://doi.org/10.1029/92WR01686.

Cordes, C. and Kinzelbach, W. (1996). Comment on 'Application of the mixed hybrid finite element approximation in a groundwater flow model: Luxury or necessity¿ by R. Mosé, P. Siegel, P. Ackerer, and G. Chavent. *Water Resources Research*, 32(6):1905–1909. https://doi.org/10.1029/96WR00567.

Crane, M. J. and Blunt, M. J. (1999). Streamline-Based Simulation of Solute Transport. *Water Resources Research*, 35(10):3061 – 3078. https://doi.org/10.1029/1999WR900145.

D'Affonseca, F. M., Finkel, M., and Cirpka, O. A. (2020). Combining implicit geological modeling, field surveys, and hydrogeological modeling to describe groundwater flow in a karst aquifer. *Hydrogeology Journal*, 28:2779 – 2802. https://doi.org/10.1007/s10040-020-02220-z.

Davis, T. A. (2004). A Column Pre-Ordering Strategy for the Unsymmetric-Pattern Multifrontal Method. *ACM Transactions on Mathematical Software (TOMS)*, 30(2):165–195. https://doi.org/10.1145/992200.992205.

Dompierre, J., Labbé, P., Vallet, M.-G., and Camarero, R. (1999). How to Subdivide Pyramids, Prisms and Hexahedra into Tetrahedra. Report cerca r99-78, École Polytechnoque Montréal: Centre de Recherche en Calcul Appliqué (CERCA). Presented at 8th International Meshing Roundtable, Lake Tahoe, California.

Durlofsky, L. J. (1994). Accuracy of mixed and control volume finite element approximations to darcy velocity and related quantities. *Water Resources Research*, 30(4):965–973. https://doi.org/10.1029/94WR00061.

Edwards, M. G. (2002). Unstructured, Control-Volume Distributed, Full-Tensor Finite-Volume Schemes with Flow Based Grids. *Computational Geosciences*, 6(3):433–452. https://doi.org/10.1023/A:1021243231313.

Edwards, M. G. and Zheng, H. (2010). Quasi-positive families of continuous Darcy-flux finite volume schemes on structured and unstructured grids. *Journal of Computational and Applied Mathematics*, 234(7):2152–2161. https://doi.org/10.1016/j.cam.2009.08.078.

Forsyth, P. (1991). A Control Volume Finite Element Approach to NAPL Groundwater Contamination. *SIAM Journal on Scientific and Statistical Computing*, 12(5):1029–1057. https://doi.org/10.1137/0912055.

Friis, H., Edwards, M., and Mykkeltveit, J. (2009). Symmetric Positive Definite Flux-Continuous Full-Tensor Finite-Volume Schemes on Unstructured Cell-Centered Triangular Grids. *SIAM Journal on Scientific Computing*, 31(2):1192–1220. https://doi.org/10.1137/070692182.

Ginn, T. R. (2001). Stochastic-convective transport with nonlinear reactions and mixing: finite streamtube ensemble formulation for multicomponent reaction systems with intra-streamtube dispersion. *Journal of Contaminant Hydrology*, 47(1-2):1 – 28. https://doi.org/10.1016/S0169-7722(00)00167-4.

Helmig, R. and Huber, R. (1998). Comparison of Galerkin-type discretization techniques for two-phase flow in heterogeneous porous media. *Advances in Water Resources*, 21(8):697–711. https://doi.org/10.1016/S0309-1708(97)00023-7.

Hoteit, H., Erhel, J., Mosé, R., Philippe, B., and Ackerer, P. (2002). Numerical Reliability for Mixed Methods Applied to Flow Problems in Porous Media. *Computational Geosciences*, 6(2):161 – 194. https://doi.org/10.1023/A:1019988901420.

Huber, R. and Helmig, R. (2000). Node-centered finite volume discretizations for the numerical simulation of multiphase flow in heterogeneous porous media. *Computational Geosciences*, 4:141–164. https://doi.org/10.1023/A:1011559916309.

Hughes, T. J. R., Engel, G., Mazzei, L., and Larson, M. G. (2000). The Continuous Galerkin Method Is Locally Conservative. *Journal of Computational Physics*, 163(2):467–488. https://doi.org/10.1006/jcph.2000.6577.

Huyakorn, P. S. and Pinder, G. F. (1983). *Computational methods in subsurface flow*. Academic Publishers, New York.

Ippisch, O. (2001). *Coupled Transport in Natural Porous Media*. PhD thesis, University of Heidelberg, Heidelberg. https://doi.org/10.11588/heidok.00001872.

Juanes, R. and Matringe, S. F. (2009). Unified Formulation for High-Order Streamline Tracing on Two-Dimensional Unstructured Grids. *Journal of Scientific Computing*, 38(1):50 – 73. https://doi.org/10.1007/s10915-008-9228-2.

Kees, C. E., Farthing, M. W., and Dawson, C. N. (2008). Locally conservative, stabilized finite element methods for variably saturated flow. *Computer Methods in Applied Mechanics and Engineering*, 197(51):4610 – 4625. https://doi.org/10.1016/j.cma.2008.06.005.

Kinzelbach, W. (1992). *Numerische Methoden zur Modellierung des Transports von Schadstoffen im Grundwasser*. Vol. 21 of Schriftenreihe gwf Wasser - Abwasser. R. Oldenbourg Verlag, München.

Kinzelbach, W. and Ackerer, P. (1986). Modelisation de la propagation d'un contaminant dans un champ d'ecoulement transitoire. *Hydrogeologie*, 2:197–205.

Kolditz, O. (2002). *Computational Methods in Environmental Fluid Mechanics*. Springer-Verlag, Berlin, Heidelberg. https://doi.org/10.1007/978-3-662-04761-3.

Larson, M. G. and Niklasson, A. J. (2004). A conservative flux for the continuous Galerkin method based on discontinuous enrichment. *CALCOLO*, 41(2):65 – 76. https://doi.org/10.1007/BF02637255.

Loschko, M., Wöhling, T., Rudolph, D. L., and Cirpka, O. A. (2016). Cumulative relative reactivity: A concept for modeling aquifer-scale reactive transport. *Water Resources Research*, 52(10):8117 – 8137. https://doi.org/10.1002/2016WR019080.

Loschko, M., Wöhling, T., Rudolph, D. L., and Cirpka, O. A. (2018). Accounting for the Decreasing Reaction Potential of Heterogeneous Aquifers in a Stochastic Framework of Aquifer-Scale Reactive Transport. *Water Resources Research*, 54(1):442–463. https://doi.org/10.1002/2017WR021645.

Martin, S., Klingler, S., Dietrich, P., Leven, C., and Cirpka, O. A. (2020). Structural controls on the hydrogeological functioning of a floodplain. *Hydrogeology Journal*, 28:2675 – 2696. https://doi.org/10.1007/s10040-020-02225-8.

Matringe, S. F., Juanes, R., and Tchelepi, H. A. (2006). Robust streamline tracing for the simulation of porous media flow on general triangular and quadrilateral grids. *Journal of Computational Physics*, 219(2):992 – 1012. https://doi.org/10.1016/j.jcp.2006.07.004.

McDonald, M. G. and Harbaugh, A. W. (1988). A modular three-dimensional Finite Difference groundwater flow model. Modelling Techniques Book 6, Scientific Software Group.

Mualem, Y. (1976). A new model for predicting the hydraulic conductivity of unsaturated porous media. *Water Resources Research*, 12(3):513 – 522. https://doi.org/10.1029/WR012i003p00513.

Nédélec, J. C. (1980). Mixed finite elements in $\mathbb{R}^3$. *Numerische Mathematik*, 35(3):315 – 341. https://doi.org/10.1007/BF01396415.

Nédélec, J. C. (1986). A new family of mixed finite elements in $\mathbb{R}^3$. *Numerische Mathematik*, 50(1):57 – 81. https://doi.org/10.1007/BF01389668.

Ngo, A. Q. T., Bastian, P., and Ippisch, O. (2015). Numerical solution of steady-state groundwater flow and solute transport problems: Discontinuous Galerkin based methods compared to the Streamline Diffusion approach. *Computer Methods in Applied Mechanics and Engineering*, 294:331–358. https://doi.org/10.1016/j.cma.2015.06.008.

Odsæter, L. H., Wheeler, M. F., Kvamsdal, T., and Larson, M. G. (2017). Postprocessing of non-conservative flux for compatibility with transport in heterogeneous media. *Computer Methods in Applied Mechanics and Engineering*, 315:799 – 830. https://doi.org/10.1016/j.cma.2016.11.018.

Panday, S., Huyakorn, P. S., Therrien, R., and Nichols, R. L. (1993). Improved three-dimensional finite-element techniques for field simulation of variably saturated flow and transport. *Journal of Contaminant Hydrology*, 12(1):3 – 33. https://doi.org/10.1016/0169-7722(93)90013-I.

Pollock, D. W. (1988). Semianalytical Computation of Path Lines for Finite-Difference Models. *Groundwater*, 26(6):743 – 750. https://doi.org/10.1111/j.1745-6584.1988.tb00425.x.

Povich, T. J., Dawson, C. N., Farthing, M. W., and Kees, C. E. (2013). Finite element methods for variable density flow and solute transport. *Computational Geosciences*, 17(3):529–549. https://doi.org/10.1007/s10596-012-9330-2.

Prévost, M., Edwards, M. G., and Blunt, M. J. (2002). Streamline Tracing on Curvilinear Structured and Unstructured Grids. *Society of Petroleum Engineers Journal*, 7(2):139 – 148. https://doi.org/10.2118/66347-MS.

Putti, M. and Cordes, C. (1998). Finite Element Approximation of the Diffusion Operator on Tetrahedra. *SIAM Journal on Scientific Computing*, 19(4):1154–1168. https://doi.org/10.1137/S1064827595290711.

Putti, M. and Sartoretto, F. (2009). Linear Galerkin vs mixed finite element 2D flow fields. *International Journal for Numerical Methods in Fluids*, 60(9):1011 – 1031. https://doi.org/10.1002/fld.1929.

Raviart, P. A. and Thomas, J. M. (1977). A mixed finite element method for 2-nd order elliptic problems. In Galligani, I. and Magenes, E., editors, *Mathematical Aspects of Finite Element Methods*, pages 292 – 315, Berlin, Heidelberg. Springer.

Richards, L. A. (1931). Capillary conduction of liquids through porous mediums. *Physics*, 1(5):318–333. https://doi.org/10.1063/1.1745010.

Richardson, L. F. (1922). *Weather prediction by numerical process*. Cambridge: Cambridge mathematical library. Cambridge University Press, New York, 2 edition. https://doi.org/10.1017/CBO9780511618291.

Rognes, M., Kirby, R., and Logg, A. (2009). Efficient Assembly of $H(\mathrm{div})$ and $H(\mathrm{curl})$ Conforming Finite Elements. *SIAM Journal on Scientific Computing*, 31(6):4130 – 4151. https://doi.org/10.1137/08073901X.

Schiavazzi, D. (2013). *Redundant Multiresolution Uncertainty Propagation*. PhD thesis, University of Padova, Padova.

Scudeler, C., Putti, M., and Paniconi, C. (2016). Mass-conservative reconstruction of galerkin velocity fields for transport simulations. *Advances in Water Resources*, 94:470 – 485. https://doi.org/10.1016/j.advwatres.2016.06.011.

Selzer, P., Allgeier, J., Therrien, R., and Cirpka, O. A. (2021). Finite-volume flux reconstruction and semi-analytical particle tracking on triangular prisms for finite-element-type models of variably-saturated flow. *Advances in Water Resources*, 154:103944. https://doi.org/10.1016/j.advwatres.2021.103944.

Selzer, P. and Cirpka, O. A. (2020). Postprocessing of standard finite element velocity fields for accurate particle tracking applied to groundwater flow. *Computational Geosciences*, 24(4):1605 – 1624. https://doi.org/10.1007/s10596-020-09969-y.

Shapiro, A. M. and Cvetkovic, V. D. (1988). Stochastic analysis of solute arrival time in heterogeneous porous media. *Water Resources Research*, 24(10):1711 – 1718. https://doi.org/10.1029/WR024i010p01711.

Shewchuk, J. R. (2005). http://www.cs.cmu.edu/~quake/triangle.html. Accessed: 30.10.2018.

Srinivasan, G. and Lipnikov, K. (2013). On the Reconstruction of Darcy Velocity in Finite-Volume Methods. *Transport in Porous Media*, 96:337 – 351. https://doi.org/10.1007/s11242-012-0091-0.

Su, D., Mayer, K. U., and MacQuarrie, K. T. B. (2020). Numerical investigation of flow instabilities using fully unstructured discretization for variably saturated flow problems. *Advances in Water Resources*, 143:103673. https://doi.org/10.1016/j.advwatres.2020.103673.

Sun, S. and Wheeler, M. F. (2006). Projections of velocity data for the compatibility with transport. *Computer Methods in Applied Mechanics and Engineering*, 195(7):653 – 673. https://doi.org/10.1016/j.cma.2005.02.011.

Tecplot Inc. (2019). *User's Manual Tecplot 360 EX 2019 Release 1*. Bellevue, WA.

Therrien, R. and Sudicky, E. A. (1996). Three-dimensional analysis of variably-saturated flow and solute transport in discretely-fractured porous media. *Journal of Contaminant Hydrology*, 23(1):1 – 44. https://doi.org/10.1016/0169-7722(95)00088-7.

Tompson, A. F. B. and Gelhar, L. W. (1990). Numerical simulation of solute transport in three-dimensional, randomly heterogeneous porous media. *Water Resources Research*, 26(10):2541–2562. https://doi.org/10.1029/WR026i010p02541.

van Genuchten, M. T. (1980). A Closed-form Equation for Predicting the Hydraulic Conductivity of Unsaturated Soils. *Soil Science Society of America Journal*, 44(5):892 – 898. https://doi.org/10.2136/sssaj1980.03615995004400050002x.

Vidotto, E., Helmig, R., Schneider, M., and Wohlmuth, B. (2018). Streamline method for resolving sharp fronts for complex two-phase flow in porous media. *Computational Geosciences*, 22:487 – 1502. https://doi.org/10.1007/s10596-018-9767-z.