INSTITUT FÜR
GEOWISSENSCHAFTEN

Dissertation
Zur Erlangung des akademischen Grades
„Doktor der Naturwissenschaften“
im Promotionsfach Geologie/Paläontologie

am Fachbereich
Chemie, Pharmazie und Geowissenschaften
der Johannes Gutenberg-Universität Mainz

Digital Twin Preparation and Hydro-mechanical Barite Deposition Modelling in Fractured Rocks

vorgelegt von
Marcel Reinhardt
geboren am 16.04.1993 in
Nastätten

JG|U
JOHANNES GUTENBERG
UNIVERSITÄT MAINZ

Mainz, Oktober 2022
Arbeitsgruppe Hydrogeochemie
Institut für Geowissenschaften
Johannes Gutenberg–Universität Mainz

Abstract

Global warming is a central challenge of this century. Thus, reducing anthropogenic greenhouse gas emissions by transitioning to renewable energy sources is critical. Geothermal plants have a considerable potential as a renewable energy source since they can constantly generate heat and energy. However, disequilibrium in circulating fluids in the (fractured) reservoir can disturbed this constant heat generation by e.g., mineral precipitation. Mineral scaling can reduce the efficiency of plants; therefore, it must be minimized to guarantee a sustainable operation.

This work aims to contribute to a deeper understanding of mineral scaling in fractured reservoirs and is part of the joint project called ReSalt. The digital rock physics (DRP) workflow was applied to model hydro-mechanical particle deposition in fractured sandstone cores, which pose as analogue rocks to reservoirs in Germany. The analysis focuses on barite particles since they are typical scaling minerals in these reservoirs. Two main problems were studied in this study. The first topic is dedicated to the extraction of reliable digital twins (DTs) of rocks. This work focuses on two crucial steps: the segmentation and the correction of the partial volume effect (PVE). Some Novel machine learning approaches were applied and compared to the conventional segmentation methods. The results highlighted the potential of the machine learning segmentation methods. Based on the segmentation results, a novel workflow was developed to correct the PVE in fractures. The fractures were reconstructed at a higher resolution and validated by laboratory data (i.e., permeability, tracer curves, mean aperture). It was demonstrated that the fracture reconstruction resulted in plausible DTs. Recommendations for future studies were made to apply and advance the presented framework.

The second topic of this study focuses on the influence of various parameters (temperature, flow rate, adhesion forces, particle size, and amount) on the hydro-mechanical barite particle deposition. A numerical sensitivity analysis was conducted with two validated reconstructed fracture models. Insights on optimal conditions for the least particle deposition were given. In the conclusion, assumptions in the analysis are discussed, and an outlook for future studies is presented.

This thesis attempts to expand the concept of digital rock analysis for fractured rocks. Novel methods were applied, and their benefits and current limitations were discussed. Since the PVE can heavily affect DTs of rock fractures, the developed reconstruction workflow might contribute to creating reliable DTs in future studies. The sensitivity analysis is the first step in understanding the importance of various parameters on the process of hydro-mechanical particle deposition in fractures. Insights from this dissertation can contribute to the design of more complex numerical and experimental studies.

Zusammenfassung

Die gegenwärtige Erderwärmung ist eine der größten Herausforderungen dieses Jahrhunderts. Daher ist die Reduzierung der anthropogenen Treibhausgasemissionen durch den Übergang zu erneuerbaren Energien unabdingbar. Geothermische Kraftwerke können hierbei einen wichtigen Beitrag leisten, da diese eine konstante Erzeugung von Wärme und Strom leisten können. Allerdings werden die Fluide während der Zirkulation aus dem chemischen Gleichgewicht gebracht. Dadurch können u.a. Minerale ausfällen, welche die Fließpfade im Reservoir verändern können. Dieses Scaling kann die Effizienz der Kraftwerke verringern und muss daher für einen nachhaltigen Betrieb minimiert werden.

Diese Dissertation leistet einen Beitrag zum Verständnis von Scaling in geklüfteten Reservoiren und ist Teil des Verbundprojekts ReSalt. Im Projekt wurden Analoggesteine zu Untergrundreservoiren in Deutschland und das hier typisch auftretende Scaling-Mineral Baryt behandelt. In dieser Arbeit wurde das Digital-Rock-Physics- Konzept (DRP) angewandt, um hydro-mechanische Partikelablagerung in geklüfteten Sandsteinproben zu modellieren. Das erste Hauptthema dieser Arbeit widmete sich der Erstellung von digitalen Zwillingen (DZ). Hierbei standen zwei Aspekte im Fokus: die Segmentierung und die Korrektur des Partial-Volume-Effects (PVE). Neuste Machine-Learning (ML) und konventionelle Methoden wurden zur Segmentierung angewandt und verglichen. Hierbei konnte das Potential der ML Ansätze demonstriert werden. Basierend auf der Segmentierung wurde ein neuer Korrekturansatz des PVE entwickelt. Die Klüfte wurden in höherer Ortsauflösung rekonstruiert und mit Labordaten (Permeabilität, Tracerkurven, Aperturen) validiert. Es konnte demonstriert werden, dass realistische DZ durch die Korrektur rekonstruiert wurden. Zusätzlich wurden Vorschläge zur Anwendung und Weiterentwicklung der Korrektur in zukünftigen Studien formuliert.

Die zweite Hauptthematik dieser Arbeit widmete sich dem Einfluss von fünf Parametern (Temperatur, Durchflussrate, Adhäsion, Partikelgröße und -anzahl) auf die hydro-mechanische Barytablagerung. Eine Parameterstudie wurde mit zwei rekonstruierten Klüften durchgeführt. So konnten Erkenntnisse hinsichtlich optimaler Bedingungen, für minimale Partikelablagerungen, gewonnen werden. Abschließend wurden die Modelle diskutiert und ein Ausblick auf zukünftige Studien gegeben werden.

Zusammenfassend versucht diese Dissertation das Konzept der DRP auf geklüfteten Proben zu erweitern. Neuste Methoden wurden angewandt und deren Vorteile und aktuelle Limitationen diskutiert. Da der PVE einen großen Einfluss auf DZ von Klüften haben kann, kann der vorgestellte Korrekturansatz in zukünftigen Studien Verwendung finden, um realistische DZ zu erstellen. Die Sensitivitätsstudie stellt den ersten Schritt im Verständnis der Relevanz von ausgewählten Parametern auf die hydro-mechanischer Partikelablagerung in Klüften dar. Erkenntnisse dieser Dissertation können dabei helfen komplexere numerische und experimentelle Studien besser zu planen.

Table of contents

Table of figures VI
List of tables VIII
Abbreviation index IX
Formula index X
1. Introduction 1
  1.1 Motivation 1
  1.2 Research project ReSalt 4
  1.3 Main objectives and thesis outline 6
2. Fundamentals 8
  2.1 Fractured media 8
  2.2 Computed Tomography & Artefacts 11
    2.2.1 Basic principles 11
    2.2.2 Artefacts 16
  2.3 Partial volume effect 19
  2.4 Segmentation 22
    2.4.1 Conventional methods 23
    2.4.2 Machine learning methods 24
  2.5 Fluid dynamics 28
  2.6 Particle transport 29
3. Materials & Methods 34
  3.1 Samples 34
  3.2 Computed Tomography 35
  3.3 Segmentation 35
    3.3.1 Conventional methods 35
    3.3.2 Convolutional neural network 38
    3.3.3 Random forest classifier 40
  3.4 PVE Correction 41
  3.5 Validation parameters 49
    3.5.1 Water injection and tracer experiments 50
    3.5.2 Barite particle migration experiments 52
    3.5.3 Fracture aperture and roughness 55
  3.6 Simulation setups 57
4. Case studies 62
  4.1 Comparison of fracture segmentation methods 62
    4.1.1 Conventional segmentation methods 62
    4.1.2 Machine learning methods 65
    4.1.3 Fracture properties 67
    4.1.4 Discussion and conclusion 72

4.2 Fracture reconstruction 74
  4.2.1 Workflow 75
  4.2.2 Results and Discussion 78
  4.2.3 Comparison to segmented fractures 87
  4.2.4 Discussion and conclusion 88
4.3 Barite particle deposition modelling 90
  4.3.1 Base model setup and validation 91
  4.3.2 Sensitivity analysis 95
  4.3.3 Discussion and conclusion 115
5. Summary and Outlook 119
Appendix 124
References 146

Table of figures

Figure 1. Illustration of different aperture definitions on a rough fracture 10
Figure 2. Illustration of the DRP workflow 13
Figure 3. Common image artefacts.. 18
Figure 4. Illustration of the partial volume effect... 20
Figure 5. Overview of commonly applied filters in the interactive machine learning workflow 27
Figure 6. The grey value histograms of the Flechtinger and Remlinger cores 36
Figure 7. The segmentation procedure is illustrated on the Rem_1_S2 37
Figure 8. Architecture of the implemented U-net CNN.. 38
Figure 9. Illustration of the principles for the applied fracture reconstruction 43
Figure 10. Flow chart of the applied reconstruction workflow.. 45
Figure 11. Workflow to reconstruct the fracture from the XCT images.. 48
Figure 12. Size distribution of the barite particles deposited in the Fle_9_S2 fracture...................... 53
Figure 13. SEM snapshots of the deposited barite particles 54
Figure 14. Workflow to calculate the true fracture apertures from a binary image stack 56
Figure 15. Comparison of the experimental flow cell and the model... 59
Figure 16. Comparison of the applied segmentation methods on the Fle_8_S1.............................. 63
Figure 17. Comparison of the applied segmentation methods on the Rem_1_S2 64
Figure 18. Results of the RF segmentation.. 66
Figure 19. Comparison of segmented structures of the Fle_8_S1 70
Figure 20. Comparison of segmented structures of the Rem_1_S2 71
Figure 21. Workflow to reconstruct the validated digitized fractures 77
Figure 22. Result of the reconstruction of the samples Fle_8_S1 and Rem_1_S2............................ 80
Figure 23. Estimation of the hydraulic aperture 84
Figure 24. Apertures of the reconstructions and segmentations of the reference samples 88
Figure 25. Setup for conducting the barite particle migration simulations 92
Figure 26. Qualitative comparison of the barite deposition results 94
Figure 27. Computed flow fields for the base models 97
Figure 28. Assessment of the numerical error of the barite migration simulations 98
Figure 29. Varying particle amount 100
Figure 30. Varying flow rate 102
Figure 31. Residence time and first arrival with a varying flow rate... 102
Figure 32. Influence of the flow rate on barite particle deposition 103
Figure 33. Varying temperature 104
Figure 34. Illustration of particle deposition with varying adhesion forces..................................... 105
Figure 35. Critical velocity with varying parameters 106
Figure 36. Varying adhesion forces 107
Figure 37. Sensitivity analysis for the particle size 108
Figure 38. Number of collisions for each particle size.. 110
Figure 39. Simulation results of injecting a constant mass of barite into the fractures 111
Figure 40. Comparison of particle and tracer transport.. 112
Figure 41. Particle deposition pattern.. 114
Figure 42. Flow velocity and particle deposition... 115
Figure 43. Python script to calculate local apparent fracture apertures with a local CT_{mat} value..... 125
Figure 44. Python script to calculate local apparent fracture apertures with a global CT_{mat} value. . 127
Figure 45. Matlab script for calculating the fracture skeleton... 127
Figure 46. Matlab script to grow a fracture skeleton according to the local apparent aperture. 130
Figure 47. Python script to measure the apertures of segmented fractures (Step0). 131
Figure 48. Python script to measure the apertures of segmented fractures (Step1). 137
Figure 49. Python script to measure the apertures of segmented fractures (Step2). 139
Figure 50. Result of the reconstruction of the samples Fle_7_S1 and Fle_8_S2 140

Figure 51. Result of the reconstruction of the samples Fle_9_S1 and Fle_9_S2 141
Figure 52. Result of the reconstruction of the samples Rem_1_S1 and Rem_1_S3 142
Figure 53. Result of the reconstruction of the samples Rem_2_S1 and Rem_2_S3 143
Figure 54. Apertures of the reconstructions and segmentations of the Remlinger samples 144
Figure 55. Apertures of the reconstructions and segmentations of the Flechtinger samples.......... 145

List of tables

Table 1. Measured flow rate for each flow-through experiment 51
Table 2. Selected equations for approximating the hydraulic aperture of the DTs 57
Table 3. Amount of training data provided to the RF classifier.. 65
Table 4. Loss and accuracy values of the CNN segmentation approach 67
Table 5. Overview of segmented fracture properties 67
Table 6. Parameters applied during the reconstruction workflow 82
Table 7. Overview of the properties of all reconstructed DTs 84
Table 8. Comparison of parameters describing simulated and real breakthrough curves 86
Table 9. Properties of the reconstructions and segmentations of the reference samples................ 87
Table 10. Summary of base model parameters and the ranges for the sensitivity analysis................ 96
Table 11. Mean particle – wall collisions in the reference samples... 103
Table 12. Comparison of the analysed systems in the ReSalt and ResKin projects. 122

Abbreviation index

3-D	Three-dimensional
BC	Boundary Condition
CCD	Charge-Coupled Device
CNN	Convolutional Neural Network
CT	Computed Tomography
CFMA	Calibration-free Missing Attenuation
DRP	Digital Rock Physics
DT	Digital Twin
EC	Electrical Conductivity
Fle	Flechtinger
FoV	Field of View
FTP	Fracture Trace Point
FWHM	Full-Width-Half-Maximum
GAD	GeoDict Analytic Data
HU	Hounsfield Units
HPTC	High-Pressure High-Temperature Column
IPSF	Inverse-Point-Spread Function
LCL	Local Cubic Law
MA	Missing Attenuation
PH	Peak Height
PVE	Partial Volume Effect
RAM	Random-access memory
ReLU	Rectified linear unit
Rem	Remlinger
ReSalt	Reaktive Reservoirsysteme – Lösung und Fällung von Salzen und die Auswirkungen auf die hydraulischen und mechanischen Gebirgseigenschaften
RF	Random Forest
RUB	Ruhr University Bochum
SEM	Scanning Electron Microscope
TUD	Technical University Darmstadt
TWS	Trainable Weka Segmentation
XCT	X-Ray Computed Tomography

Formula index

Symbol	SI Unit	Name
a_0	m	Adhesion distance
a_h	m	Hydraulic aperture
a_m	m	Mechanical aperture
b_i	m	Position of background pixel i
BCE	dimensionless	Binary cross entropy
c	mol m^{-3}	Particle concentration
CT_i	dimensionless	CT value of a pixel/voxel
CT_{mat}	dimensionless	Average CT value of the matrix
CT_{air}	dimensionless	Average CT value of air
C_c	dimensionless	Cunningham correction factor
γ	kg s^{-1}	Friction coefficient
d	m	Local apparent fracture aperture, calculated with CFMA
D	m^2 s	Molecular diffusion coefficient
E	eV	Energy of X-ray beam
$\vec{f}$	N	Force density field
g	m s^{-2}	Gravitational acceleration
g_t	dimensionless	Ground truth
H	J	Hamaker constant
dh	m	Hydraulic pressure gradient
I	number of photons s^{-1} m^{-2}	X-ray beam intensity, attenuated
I_0	number of photons s^{-1} m^{-2}	X-ray beam intensity, unattenuated
I	dimensionless	Hydraulic gradient
IoU	dimensionless	Intersection over union
J_l	dimensionless	Jaccard distance loss
K	m^2	Permeability
k_B	J K^{-1}	Boltzmann constant
L	dimensionless	Loss function
L_w	m	Fracture width
L_L	m	Fracture length

Symbol	**SI Unit**	**Name**
l	m	Characteristic length scale
M	dimensionless	Geometrical magnification
m	kg	Mass
N_{vox}	dimensionless	Number of voxels that are used for the local aperture calculation after Huo et al. 2016
μ_0	m^{-1}	Attenuation coefficient of a material
μ_i	m^{-1}	Attenuation coefficient of material i
μ	Pa s = kg m^{-1} s^{-1}	Dynamic fluid viscosity
p	Pa = kg m^{-1} s^{-2}	Pressure
Pe	dimensionless	Péclet number
px_i	m	Position of fracture pixel i
φ	dimensionless	Porosity
Q	m^{3} s^{-1}	Flow rate
R	m	Spatial resolution
R_d	m	Resolution of the detector
r	m^{2}	Radius
Re	dimensionless	Reynolds number
ρ	kg m^{-3}	Fluid density
s	m	Focal spot size of the X-ray source
s_p	dimensionless	Predicted segmentation
σ_{a_m}	m	Standard deviation of the mechanical aperture
T	K	Temperature
t	s	Time
$\vec{u}$	m s^{-1}	Fluid flow velocity field
v	m s^{-1}	Typical flow velocity for the calculation of Re
$\vec{v}$	m s^{-1}	Molecular particle motion
w	dimensionless	Weighting factor
$\vec{W}(t)$	dimensionless	3-D Wiener measure
x	m	Distance X-ray beam travelled through a material
x_i	m	Distance X-ray beam travelled through material i
x_p	m	Distance travelled by a particle

1. Introduction

1.1 Motivation

The mean global surface temperature increased by around 1°C from the pre-industrial period until the last decade (2010 – 2019, with a reference period of 1850 – 1900, IPCC 2021). This ongoing global warming can be mainly attributed to the anthropogenic emission of greenhouse gases (GHG; IPCC 2021). The most important GHGs are carbon dioxide (CO_2), methane (CH_4) and nitrous oxide (N_2O), whose concentrations in the atmosphere are rising along with the global surface temperature. Sources of GHGs include the combustion of fossil fuels (oil, gas, coal), industrial processes, agriculture, biomass and biofuel burning. The consequences of this rapid global warming are severe and include sea level rising, ocean acidification, changes in ocean circulation, glacier retreating, reduction in permafrost, higher frequency of climate extremes (e.g., droughts, heatwaves, heavy precipitation, floods), and changes in precipitation and wind patterns. The degree and timescale of these effects depend on the future emission of GHGs (IPCC 2021).

Global warming has major impacts on biodiversity and humankind (IPBES 2019; IPCC 2021). Therefore, it is considered one of the biggest challenges of this century (United Nations 2015). To reduce the impact of global warming, the United Nations consented in the Paris treaty to a maximum temperature rise of around 1.5 - 2°C (UNFCCC 2015; IPCC 2021). There is a direct relationship between the cumulative CO_2 (the most important GHG) emissions and the rise in global surface temperature (IPCC 2021). Hence, limiting this temperature increase corresponds to a budget of remaining CO_2 that can be emitted. In order to reach this goal, GHG emissions must be reduced until net-zero anthropogenic CO_2 emissions are reached in the middle of this century (IPCC 2021).

Energy and heat production is one of the primary sources of GHG emissions and, therefore, a fundamental target for GHG emission reduction (IPCC 2014, BMU 2021). The transition from classical (i.e., fossil fuel-based) energy sources to renewable energy sources (e.g., photovoltaic, wind power, hydropower, tidal power, geothermal energy) is a crucial step in reaching the necessary climate goals. For example, Germany's fraction of renewable energy sources used in energy and heat production has been rising since 1990 and included in 2020 about 44 % of the gross electricity produced and 15.6 % of the heat used (BMU 2021; BMWi 2021). While the fraction of geothermal energy, as a renewable energy source, was still low in 2020 (around 0.1 % of the gross electricity production and approximately 1.5 % of the heat produced from renewable energies), it is increasing and, due to its advantages, considered as substantial for the transition to net-zero anthropogenic CO_2 emissions (Eyerer et al. 2020; BMWi 2021). Hence, it is planned to extend the employment of geothermal power for the next decades in Germany (Klaus et al. 2010; BMWi 2021; SPD, Bündnis 90/die Grünen and FDP 2021; Acksel et al. 2022).

Geothermal power has a significant potential as a sustainable energy source because of its low environmental impact, technical feasibility, constant energy production (weather independent), the magnitude of potential energy sources, and their worldwide distribution (Barbier 2002; Stefansson 2005; Stober and Bucher 2014; Li et al. 2015; Shortall et al. 2015; Moya et al. 2018). In recent years, the use of geothermal energy increased globally for both direct use (e.g., heat pumps, space heating, bathing, and swimming) and for the generation of electricity. In 2019, approximately 88 countries directly used geothermal energy, mainly through heating pumps, and about 30 countries generated electricity through geothermal energy (Huttrer 2020; Lund and Toth 2021). While shallow geothermal use (e.g. by downhole heat exchangers combined with heat pumps) is widely available, the feasibility of geothermal electricity generation depends more on regional geological conditions, e.g., the increase in temperature with depth (geothermal gradient, Stober and Bucher 2014; Shortall et al. 2015; Eyerer et al., 2017). In Germany, the majority of utilized geothermal energy is used directly (e.g., by an increasing number of installed heat pumps), and there are geologically suitable regions for geothermal electricity generation, mainly in the Upper Rhine Graben, North German Basin and South German Molasse Basin (Agemar et al. 2014; Stober and Bucher 2014; Eyerer et al. 2020; Lund and Toth 2021).

Deep geothermal systems can be classified as hydrothermal (e.g., doublet system) or petrothermal (e.g., enhanced geothermal systems, EGS). Whereas the former systems utilize the energy stored in water from aquifers, the latter systems utilize fluids to extract the energy stored in hot rocks. High permeabilities are required to extract subsurface energies efficiently. If a reservoir does not have the sufficient permeability, it can be enhanced by e.g., inducing or augmenting fractures by injecting water at high pressures. Therefore, fractures are essential pathways in these stimulated reservoirs (Zimmermann et al. 2010).

In both systems, it is common to use reservoir fluids or injected geothermal fluids in a closed loop: hot fluids are conveyed through a production well to the surface, where the energy is extracted with heat exchangers. By an injection well, the colder fluid is injected back into the subsurface, where it can flow towards the production well (while extracting energy from the subsurface) and thereby closing the loop (Stober and Bucher 2014). The reinjection is advantageous since the produced water (which can have a high content of minerals and gases) is directly and adequately disposed of, subsurface pressure is controlled, and the subsurface water is directly recharged (Stober and Bucher 2014; Kamila et al. 2021). The properties of these saline fluids depend on the characteristics of the reservoirs and the conditions during the circulation (i.e., pressure, temperature, chemical composition). Problems occurring during circulation include corrosion, subsidence, induced seismicity, and mineral precipitation (scaling) because of oversaturation with changes in pressure, temperature, and fluid composition. Therefore, a sustainable operation of geothermal plants requires careful planning of the circulation process by considering the local geological conditions, kinetics and chemical equilibrium of the circulating fluid (Stober and Bucher 2014; Kamila et al. 2021; Schäffer et al. 2022; Zhao et al. 2022).

Common examples of scaling minerals are carbonates ($CaCO_3$), sulphates (e.g., $BaSO_4$, $SrSO_4$), sulphites (e.g., FeS, PbS, CuS), or quartz (SiO_2). Scaling can form above the surface and e.g., reduce the inner diameter of tubes or lower the efficiency of heat exchangers by building an insulating layer. Precipitated mineral particles can also be transported with the reinjected fluid back into the reservoir and alter the fluid pathways by deposition in pores or fractures. Hence, scaling can influence the efficiency of the power plant (Baticci et al. 2010; Frick et al. 2011). Common remediations include special coatings, pressurizing of water at the surface, adjusting operational conditions (e.g., degree of fluid cooling, flow rate) or the addition of inhibitors or acids. When scales have formed, they can be removed by physical or chemical methods (Saadat et al. 2010; Scheiber et al. 2012; Stober and Bucher 2014; Kamila et al. 2021; Tranter et al. 2021a, b). Although anti-scaling methods are intensively studied, there is still ongoing research to find environmental friendly methods with low economic costs that work effectively against commonly encountered scaling minerals (Kioka and Nakagawa 2021; Zhao et al. 2022).

In recent years, the increase in computational power and advances in interdisciplinary sciences (like computational science or water resource research) enabled DRP to become an essential tool for analysing unique or sensitive samples (Andrä et al. 2013a, b; Ponomarev et al. 2022). The DRP workflow is usually non-destructive and allows a quantitative analysis of complex samples. Furthermore, DRP allows to carry out complex simulations on realistic three-dimensional (3-D) DT models, which can complement or even replace expansive, time-intensive laboratory experiments (Saxena et al. 2017a). Because of these advantages, DRP was applied in numerous studies to analyse e.g., sample properties, reactive flow, tight gas production or particle transport through fractures and the pore space (Sell et al. 2013; Stoll et al. 2019; Hinz et al. 2019; Zhang et al. 2019; Hale et al. 2022).

The DRP workflow can be summarized as “image and compute” (Sadeghnejad et al. 2021). The first part of this paradigm includes the imaging and digitization steps: the sample acquirement, setting up the experimental and imaging environment, reconstruction of acquired images, filtering, and segmentation (Andrä et al. 2013a, b). Recent advances in 3-D printing also allow sample replication (based on segmented images) and the creation of artificial samples with predefined properties, which also can be analysed in the DRP workflow (Kong et al. 2021). The segmented images are combined to acquire a 3-D DT of the investigated sample. The second part of the DRP paradigm includes approaches for analysing this DT (to obtain sample properties like permeability) and conducting simulations (e.g., particle transport; Andrä et al. 2013a, b). Each of these steps has its challenges and can influence the results of the analysis (Saxena et al. 2019). Hence, various approaches exist for each step, and there are numerous possibilities to combine methods in the DRP workflow for an appropriate analysis of a given sample (e.g., Iassonov et al. 2009; Schlüter et al. 2014).

In this thesis, the DRP workflow was applied to analyse barite deposition in fractured rocks. For this, medical CT scans of single fractured sandstones were processed to obtain reliable DTs. Flow and particle transport simulations were conducted with these DTs and validated with laboratory experiments. A sensitivity study was conducted to analyse the influence of changing conditions on particle deposition. The first part of this thesis presents the workflow to create meaningful DTs. Several segmentation methods were applied, and the advantages of recently developed machine learning approaches are discussed. The resulting analysis revealed the most suitable segmentation method for analysing the fractured samples. In the next step, the segmented images were corrected for the PVE by using a novel reconstruction method. Furthermore, this section discusses the influence of the PVE, highlights the importance of a meaningful validation and confers limitations of the proposed reconstruction method. In the second part, the validated DTs were used in a sensitivity study to analyse the barite deposition under varying conditions. Operational conditions were investigated (e.g., pump rate) to find optimal conditions to avoid barite deposition in fractures, minimize reservoir scaling and extend the sustainable operation of geothermal power plants.

1.2 Research project ReSalt

Mineral scaling is a common phenomenon in geothermal power plants and influences their sustainable operation (Baticci et al. 2010; Frick et al. 2011). Therefore, a deeper understanding of the relevant processes is crucial (Schreiber et al. 2016). Scaling above the surface (e.g., at the heat exchanger or in pipes) was extensively studied; however, fewer studies analysed scaling inside reservoirs (Schreiber et al. 2016; Tranter et al. 2021a, b; Zhang et al. 2022).

Barite ($BaSO_4$) is a common scaling mineral in geothermal power plants at the upper rhine graben (e.g., in Soultz-sous-Forêts) and the North German Basin (e.g., in Groß Schönebeck), which are two of the most important locations for the production of geothermal energy in Germany (Agemar et al. 2014; Nitschke et al. 2014; Regenspurg et al. 2015; Haas-Nüesch et al. 2018; Ledésert et al. 2021). The risk of barite scaling is high with highly saline fluids and high temperature or pressure changes during fluid circulation (Tranter et al. 2020). Due to the presence of radionuclides in the barite scaling, particular caution has to be taken when workers come in contact with scaled equipment (e.g., the heat exchanger or pipes) and when the scales are disposed of (Scheiber et al. 2012; Ledésert et al. 2021). Furthermore, when the low-soluble barite scale has formed, it is hard to remove (even when using acid), and commonly applied mechanical removal is expensive and time-consuming (Akyol et al. 2014). Barite scaling is especially troubling when the reservoir rocks are affected and fluid pathways (pore space or fractures) are clogged (Birner et al. 2015). Therefore, it is favourable to prevent the formation of barite scaling (Akyol et al. 2014). There is still ongoing research to inhibit barite scaling in an effective, low-cost and environmental-friendly way (Lu et al. 2019). Another approach to minimize barite scaling is to find the optimal operational parameters (e.g., pump rate) of the geothermal power plant to reduce barite precipitation (Tranter et al. 2021a, b).

The goal of the ReSalt (“Reaktive Reservoirsysteme – Lösung und Fällung von Salzen und die Auswirkungen auf die hydraulischen und mechanischen Gebirgseigenschaften“ (Reactive Reservoirsystems – Scaling and Erosion and its Impact on Hydraulic and Mechanic Reservoirproperties)) project was to provide deeper insights into reservoir scaling, by focusing on barite scaling in single fractures in reservoir rocks. Various aspects were investigated, including the kinetics of barite precipitation, the influence of surrounding conditions on precipitate morphology, the impact of scaling on reservoir properties (i.e., permeability, porosity), hydro-mechanical properties of analogue samples to sandstone reservoir rocks used in Germany, and the upscaling from the pore- to the reservoir scale. Details regarding the different investigations can be found on the project website: https://www.geo.tu-darmstadt.de/resalt/projekt_resalt/index.de.jsp.

A high-pressure-temperature column (HPTC) was built to conduct flow-through experiments in the laboratory over several weeks and under similar conditions as expected in geothermal power plants (e.g., elevated pressure and temperature, recirculating saline fluid, cooling of the fluid). This apparatus allows to simulate the complex interaction of the fractured reservoir rock with the saline fluid under conditions prevailing in a geothermal power plant (e.g., fluid circulation with a defined flow rate, changing pressure and temperature). In the conducted experiments, barite scaling was induced by directly injecting barite particles into the samples. The experiments were accompanied by medical XCT-based (X-Ray computed tomography) analysis before and SEM-based (scanning electron microscope) analysis after. This thesis aims to provide a framework to obtain a meaningful (i.e., well-validated) DT of the real samples and to simulate barite particle migration experiments to gain a deeper understanding of the deposition of barite particles in single fractures. While laboratory experiments are needed for the validation, the simulation of DT experiments can be used to conduct additional analysis more efficiently, i.e., a faster, more cost-efficient execution of numerous simulations of experiments with varying parameters. A fundamental understanding of the processes leading to particle accumulation in single fractures assists in finding optimal operational parameters to reduce scaling and allows for a more sustainable use of geothermal power plants.

The work presented in this thesis was conducted in close collaboration with the project partners, which provided the sample material and data for verifying the modelling results. The fractured cores were provided by the partners at the Technical University in Darmstadt (TUD[redacted]). At the Ruhr University in Bochum (RUB), various laboratory experiments were conducted to verify the digital models obtained from the XCT measurements. These experiments included the determination of hydraulic parameters (permeability, hydraulic aperture, fracture volume), tracer experiments (breakthrough curves), the preparation of barite particles and the analysis of their size and morphology under varying conditions (e.g., temperature, chemical composition) and flow-through experiments to observe barite deposition in the fractured cores. The barite deposition was further analysed at the RUB (by Philipp Zuber) and TUD (by Prof. Dr. Martin Ebert) in terms of deposition patterns, particle sizes and morphology.

1.3 Main objectives and thesis outline

The work presented in this thesis focused on three main objectives in the context of the ReSalt project:

(i) The segmentation process is a critical step in the DRP workflow, and there is still ongoing research and discussion about suitable methods. In recent years, machine learning approaches have been significantly improved and now allow to produce superior segmentation results compared to conventional methods. The first objective of this thesis focused on a comparison of different segmentation approaches (including conventional and machine learning methods) to determine the most suitable method for processing the acquired medical XCT images. The advantages and disadvantages of these methods were discussed, in addition to the future potential of convolutional neural networks (CNN).

(ii) The second objective focused on the influence of the PVE on the measured medical XCT images and the importance of validating the obtained DTs. In order to compensate for the PVE, a novel workflow was developed to reconstruct single fractures at higher resolution without additional calibration data. The DTs were validated with various experimental data and compared to the segmented fractures.

(iii) The final objective was devoted to the hydro-mechanical barite deposition in the digitized fractured cores. After validating the DTs, particle transport simulations were conducted under varying conditions. The influence of changing conditions (temperature, flow velocity and rate, particle size, adhesion forces, number of injected particles) on the deposition pattern and optimal conditions (i.e., where the least particles deposit) are discussed. Due to simplifications applied in the model and the lack of quantitative laboratory data, the sensitivity analysis provides only qualitative recommendations for the sustainability of fractured reservoirs. The limitations of the study are discussed, and possible extensions of the presented study are illustrated in the outlook.

The first two goals were dedicated to the preparation of a robust DT. The results and insights from the analysis and validation can be used as a guideline for conducting future studies where similar samples (single fractured rocks) are used. The third goal was to provide first insights into the particle deposition pattern of barite particles under changing conditions. To validate these insights, more extensive studies are needed e.g., with the consideration of particle morphology in the simulations. Furthermore, the insights from particle deposition in single fractures can aid in understanding scaling in fracture networks.

This thesis is organized as follows. The second chapter is devoted to the fundamental scientific principles which are the basis for the conducted study and presents the state of the art of the applied methods. The third chapter presents the utilized rock samples and details regarding the applied methods for each step in the DRP workflow: scanning (medical XCT), segmentation, PVE correction, aperture calculation and simulations (fluid flow, particle migration). Furthermore, the conducted laboratory experiments are described. The data obtained from these experiments was used to validate the DTs. The fourth chapter is divided into three parts, where each section presents the results and discussion of one primary objective: the comparison of segmentation methods (published in Reinhardt et al. 2022), the reconstruction workflow, and the barite particle deposition. The final chapter presents the conclusions, summarizes the main results, and gives an outlook for future studies.

2. Fundamentals

This chapter presents the fundamental science on which this thesis is based. The structure follows the DRP workflow, starting with the sample material. Since single fractured rocks are considered in this study, the first section focuses on fractured rocks. The next section illustrates the principles of computed tomography and imaging artefacts, with a focus on the partial volume effect in the third section. The acquired images are segmented in the next step, which is outlined in the fourth section. The segmented images are combined to create a 3-D model of the investigated samples, which is used for further analysis and simulations. The last sections present physical principles of fluid flow and particle transport and how these are implemented in simulation software.

2.1 Fractured media

In order to quantify the fluid flow and the corresponding particle migration through fractured media, a thorough characterization of the rock, fractures (single or network), flow properties and the surrounding conditions is needed (Zimmerman and Bodvarsson 1996; Dang et al. 2019; Hu et al. 2020; Liu et al. 2020; Tan et al. 2020). These (interacting) parameters include mechanical & elastic rock properties, stress conditions (and history), surface roughness, fracture wall offset, contact areas, anisotropy, the local distance between the fracture surfaces (i.e., the aperture), fracture size, fracture type (i.e., natural or artificial), flow direction, flow rate, turbulence, tortuosity, matrix permeability, interactions between the fluid and the rock (Bandis et al. 1983; Raven and Gale 1985; Zimmerman and Bodvarsson 1996; Barton and de Quadros 1997; Ge 1997; Lough et al. 1998; Boutt et al. 2006; Kling et al. 2016; Milsch et al. 2016; Kluge et al. 2017; Lang et al. 2018; Frank et al. 2020a; Cheng and Milsch 2021; Noiriel and Soulaine 2021; Zhang et al. 2022; Steefel and Hu 2022).

Various methods exist to characterize irregular fracture surfaces (i.e., roughness) qualitatively and quantitively (Magsipoc et al. 2020). These include the joint roughness coefficient (Barton 1973; Li and Zhang 2015), the relative roughness (Zimmerman et al. 1991) and statistical methods like Z2 (Myers 1962) and R_p (El-Soudani 1978). Furthermore, fractal theory can be applied to describe roughness and its dependence on the observation scale (Fardin et al. 2001; Renard et al. 2006; Candela et al. 2009). Fracture surfaces tend to be well correlated at large length scales and follow self-affine models below a certain threshold (Brown 1995). For self-affine fractures, the Hurst exponent (and the fractal dimension) can be used to express roughness (rougher surfaces exhibit lower Hurst exponents and higher fractal dimensions), roughness anisotropy and scaling behaviour (Issa et al. 2003; Candela et al. 2012; Ficker 2017; Abe and Deckert 2021; Gutjahr et al. 2021).

In order to derive hydraulic properties, fractures are commonly described based on their mechanical aperture, which represents the distance between the opposing fracture walls, measured perpendicular to a reference plane (Olsson and Barton 2001). Usually, the mechanical aperture is calculated as the arithmetic mean of these local distances (Renshaw 1995). A famous method to relate this mechanical aperture to the hydraulic properties of a fracture (flow rate and permeability, the potential of fluids to flow through the fracture) is the cubic law. This relationship is derived from the Navier-Stokes equation (see section 2.5) and assumes two smooth parallel plates, which are separated by a constant fracture aperture and with laminar flow conditions (Witherspoon et al. 1980; Zimmerman and Yeo 2000). Deviations of the hydraulic properties calculated by the cubic law from measurements of fractures are reported in the literature and can be related to the complexities of actual fractures (e.g., roughness, aperture distribution, contact areas). The influence of these parameters on the deviation depends on individual fractures. For example, Kluge et al. (2017) reported a diminishing influence of fracture roughness with larger apertures. Numerous modifications to the cubic law exist to improve its validity (e.g., Zimmerman et al. 1991; Zimmerman and Bodvarsson 1996; Konzuk and Kueper 2004; Li et al. 2008a; Kluge et al. 2017, Blöcher et al. 2019; He et al. 2021).

As a first adaption, the mechanical aperture is replaced by the effective hydraulic aperture, which relates the actual fluid flow through a theoretical parallel plate fracture (with a constant distance equal to the hydraulic aperture) that would support this fluid flow (Figure 1). The hydraulic aperture can be measured directly (e.g., Cheng et al. 2020; Hale et al. 2020) or indirectly by first determining the mechanical aperture and then applying conversion formulas. The mechanical aperture can be measured by numerous methods, including XCT (e.g., Kling et al. 2016), profilometry (e.g., Matsuki et al. 1999), photogrammetry (e.g., Zambrano et al. 2019b) or nuclear magnetic resonance imaging (e.g., Renshaw et al. 2000). Several correlations exist to approximate the hydraulic aperture from the mechanical aperture (e.g., Renshaw 1995; Barton and de Quadros 1997; Matsuki et al. 1999; Xiong et al. 2011; Xie et al. 2015; Kling et al. 2017; Tan et al. 2020, see Table 2 in subsection 3.5.3); however, due to limitations in these approaches (e.g., only a limited number of fracture parameters is considered), a universally accepted formulation does not exist (Souley et al. 2015; Kling et al. 2017). Usually, the hydraulic aperture is smaller than the mechanical aperture (Renshaw 1995; Xiong et al. 2011; Xie et al. 2015; Kling et al. 2017).

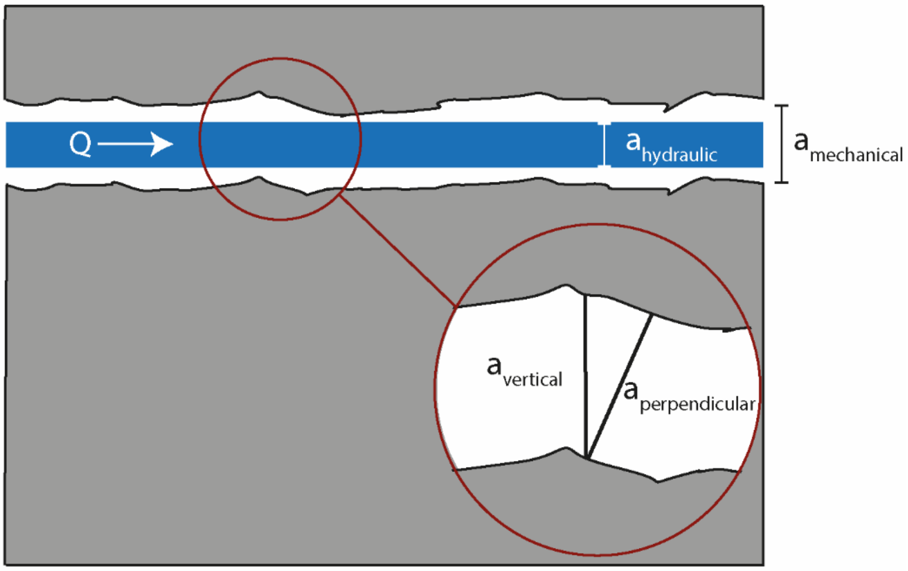


Figure 1. Illustration of different aperture definitions on a rough fracture
The mechanical aperture ($a_{mechanical}$) can be expressed as the arithmetic mean of the local distances between the fracture surfaces perpendicular to a reference plane (e.g., a fitted plane through a rough fracture surface, Vogler et al. 2018*; Renshaw 1995). In the cubic law assumption, the real fracture is simplified to two smooth parallel plates with a constant distance equal to the mechanical aperture. The effective hydraulic aperture ($a_{hydraulic}$) relates the fluid flow rate (Q) in an actual fracture with a parallel plate model that would support the same flow rate. The hydraulic aperture is the distance of these parallel plates. The local apertures can also be determined vertically ($a_{vertical}$, also called apparent aperture) or perpendicular to the fracture surfaces ($a_{perpendicular}$, or the true aperture).*

In addition to the application of a hydraulic aperture, more complex modifications of the cubic law were developed over the last decades. These approaches can be classified into two groups (He et al. 2021). Modifications of the first group introduce a correction factor to account for one or more fracture properties like surface roughness (e.g., Witherspoon et al. 1980; Kottwitz et al. 2020), friction (considering the Reynolds number, e.g., Zhang et al. 2015), or tortuosity (also considering contact area, e.g., Cook et al. 1990; Xiao et al. 2013). Furthermore, several averaging methods can be applied to determine a suitable mean aperture value (e.g., Neuzil and Tracy 1981; Tsang and Witherspoon 1981; Renshaw 1995; Konzuk and Kueper 2004). Modifications of the second group are based on the local cubic law (LCL) assumption. Since the cubic law assumes a fixed aperture value, its application is limited when the fracture apertures are changing. The idea of the LCL approach is to divide the fracture into smaller segments where the aperture is assumed to be approximately constant (i.e., the cubic law is considered to be locally valid for each segment). The cubic law is then applied over each segment to get an improved estimate of the hydraulic properties of the fracture (Brush and Thomson 2003; Wang et al. 2015; Ju et al. 2019; He et al. 2021). However, approaches from both groups still have limited validity because of e.g., negligence of local-scale flow behaviours (for approximations in the first group, especially for complex fractures with significant aperture variations and high roughness, He et al.

2021), sudden aperture variations (for LCL approaches, Brush and Thomson 2003) and high computational demands (for LCL approaches, He et al. 2021). Deviations of the cubic law formulations are reported in many theoretical (e.g. Zimmerman and Yeo 2000), numerical (e.g., Brush and Thomson 2003; Wang et al. 2015; Kottwitz et al. 2020) and experimental studies (e.g., Yeo et al. 1998; Konzuk and Kueper 2004; Chen et al. 2017). An extensive summary of the various cubic law variations and limitations can be found in He et al. (2021).

2.2 Computed Tomography & Artefacts

The key paradigms of digital rock analysis (digital rock physics, chemistry or biology) are often expressed as “image and compute” (Andrä et al. 2013a; Sadeghnejad et al. 2021). Thereby, these workflows heavily depend on the accurate imaging of the sample material. Various imaging techniques can be applied for this investigation workflow: e.g., computed tomography based on X-ray- (with the particular case of synchrotron-based XCT), neutron- or positron-radiation, or microscopy (with a camera) in combination with microfluidic studies (e.g., Kulenkampff et al. 2008; Zambrano et al. 2019a; Hassannayebi et al. 2021; Jacob et al. 2021; Sadeghnejad et al. 2021). This section focuses on X-ray computed tomography since this method is commonly used in DRP and was applied in this thesis.

2.2.1 Basic principles

Electromagnetic radiation with a wavelength in the range of pm – nm is called X-rays. Corresponding to the small wavelength, X-rays possess high energies (Wallace and Hobbs 2006), which allow them to penetrate objects that are impermeable to visible light (Clausnitzer and Hopmans 2000). During XCT measurements, many 2-D images (composed of pixels) are created and afterwards combined into a 3-D image (composed of voxels). Therefore, XCT provides (usually) non-destructive 3-D information about the internal structures of nm- to m-sized samples and offers the possibility of time-resolved information (4-D). Because of these advantages, XCT is applied in many scientific fields, including material science, food science, metrology, earth sciences and biomedical sciences (Withers et al. 2021). Depending on the achievable resolution (and therefore the measurable sample size), lab-based XCT can be categorized into medical or industrial CT (with voxel sizes ≥ 100 µm), micro- (voxel sizes ≥ 0.1 µm) and nano-CT (with voxel sizes down to ≈ 10 nm, Withers et al. 2021). This subsection covers the main principles of XCT: the experimental setup, the generation of radiation, the interaction with materials, image reconstruction, and artefacts that can occur in the reconstructed images. Since medical XCT was applied in this thesis, the following paragraphs focus on conventional lab-based XCT.

A brief summary of the advantages of synchrotron-based XCT can be found at the end of this subsection.

In a XCT setup, X-rays are generated at a source. The beam penetrates a sample, interacts with it and changes with respect to the intensity and phase. The altered beam exits the sample and hits a detector, which measures the 2-D beam intensity. The result is a 2-D projection at a given angle. In the next step, the angle between the sample and the source and detector is changed (i.e., either the sample rotates, and the source and detector remain static or vice versa), and another 2-D projection is generated. This process is repeated until usually about 100 – 3600 projections (at different angles) are acquired. The projections are reconstructed as 2-D greyscale (or CT numbers) cross-sections of the object. These cross-sections can be combined to obtain a 3-D greyscale object (Withers et al. 2021). In order to account for background noise and variations in pixel response of detectors, the offset (detector readings when the X-ray source is turned off) and gain (detector readings with an active X-ray source and no object between source and detector) calibrations are conducted before measurements (Ketcham and Carlson 2001).

The main components of an experimental setup are the X-ray source, sample stage and detector (Figure 2). Depending on the experimental requirements, varying realisations of the basic setup exist e.g., either the sample or the X-ray source and detector (e.g., when imaging a living specimen) are rotating; the X-ray beam can be fan- or cone-shaped; or the setup is extended with lenses, gratings or for complex in-situ measurements (Clausnitzer and Hopmans 2000; Watanabe et al. 2011; Bultreys et al. 2016; Withers et al. 2021; Noiriel and Renard 2022).

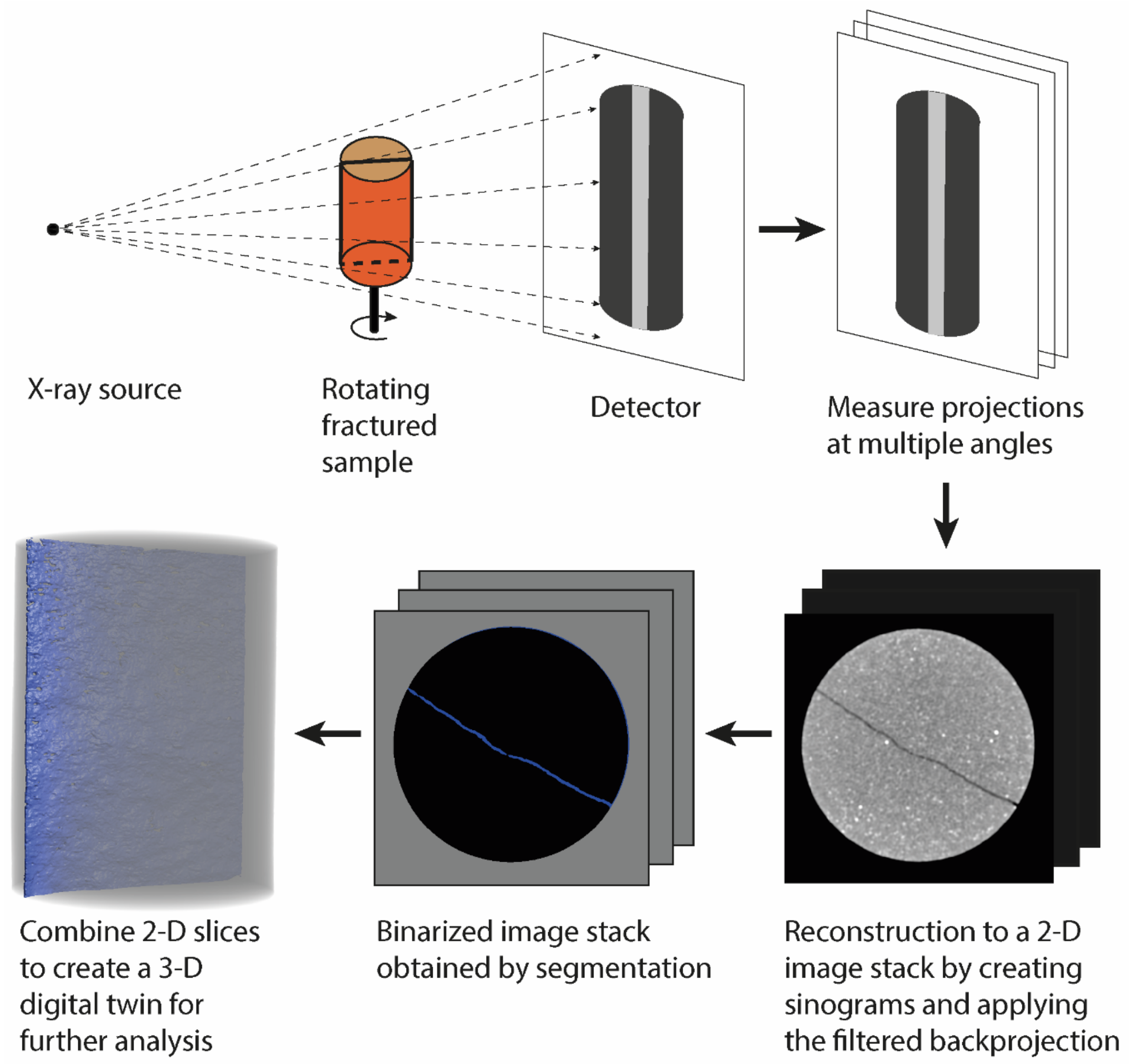

X-ray source
Rotating fractured sample
Detector
Measure projections at multiple angles
Combine 2-D slices to create a 3-D digital twin for further analysis
Binarized image stack obtained by segmentation
Reconstruction to a 2-D image stack by creating sinograms and applying the filtered backprojection

Figure 2. Illustration of the DRP workflow
Above, a typical lab-based XCT setup is shown, with a fixed source of X-rays, a rotating (fractured) sample and a fixed detector. The detector registers projections for each angle. These are summarized in sinograms, which are used for the reconstruction (e.g., by the filtered back-projection) to obtain a 2-D grey value image stack. Afterwards, the images are segmented and combined to create a 3-D DT.

Usually, the X-ray source of lab-based XCT devices is an X-ray tube, where electrons are fired at a target material (e.g., tungsten). Two processes mainly contribute to the emission of X-ray photons. The first is the deceleration of electrons, which results in bremsstrahlung. Kinetic energy is converted into electromagnetic radiation, where the amount of energy converted dictates the energy of corresponding photons. Therefore, the bremsstrahlung produces a continuous spectrum. The second source of emission is the excitation of electrons in a target material when they are struck by incoming electrons. The excited electrons are shortly elevated to a higher energy state, and when they fall back to their ground state, the difference in energy between these states is emitted as radiation. Thus, corresponding photons possess certain energies, depending on the target material. The resulting polychromatic X-ray beam is a combination of the continuous energy spectrum of the bremsstrahlung and the distinct peaks from the excitation radiation (Clausnitzer and Hopmans 2000; Wildenschild et al. 2002).

The X-ray beam has an intensity (or photon flux density) and phase that can change when it penetrates an object. The phase information can be used to obtain additional information about the penetrated objects, e.g., to enhance the contrast between different materials (Bravin et al. 2013). The intensity of an emitted monochromatic (single energy) beam I_0 is attenuated by a homogenous material according to Beer-Lambert's law:

$$I = I_o e^{-\mu_0 x} \tag{1}$$

with the beam's intensity (I) that is measured at the detector and the distance the beam travelled through the object (x). The attenuation coefficient (μ_0) of a material depends on the density and the atomic number of the material, and the X-ray energy of the beam. The attenuation of the beam can be attributed to three main processes. The first is photoelectric absorption, which occurs when an incoming photon transfers its whole energy to an inner electron of the target atom, ejecting it as a consequence. Compton scattering occurs when a photon interacts with an outer electron. The electron is ejected, whereas the photon is deflected in a different direction and only loses a part of its energy. The third process is pair production which happens when a photon is transformed into a positron-electron pair when interacting with a nucleus. Usually, for geological samples, the first two mechanisms dominate, where the importance of these processes depends on the applied X-ray energies: photoelectric absorption is the dominant effect at energies between 50 – 100 keV, Compton scattering dominates at energies up to 5 – 10 MeV and pair production at higher energies. Usually, for geological samples, beam energies are not high enough for pair production to be relevant (Ketcham and Carlson 2001). These processes also depend on the material density, and atomic weight and number (Z) of the atom struck by the photons. Photoelectric absorption is proportional to $Z^{4\text{-}5}$ and Compton scattering to Z. The change of the attenuation coefficient with changing energy is smooth, except for sharp increases at characteristic absorption edges (Ketcham and Carlson 2001).

In the case of heterogeneous samples, the contribution of each material is considered in the following adaption of the Beer-Lambert law:

$$I = I_o e^{-\sum_i \mu_i x_i} \tag{2}$$

The subscript (i) denotes the contribution of each material passed by the X-ray beam. For a polychromatic beam, the dependence of the attenuation on the beam energy (E) is considered:

$$I = \int I_o(E)[\textstyle\sum_i e^{-\mu_i(E)x_i}]dE \tag{3}$$

Since the precise spectrum of the X-ray beam is hard to quantify, and spatial variations (in the case of a fan- or cone-shaped beam) must be considered, most reconstruction methods solve equation 2, where each voxel is considered homogeneous with a single attenuation value. A cone-shaped beam is used to exploit as many photons from the source as possible (Ketcham and Carlson 2001; Wildenschild et al. 2002; Ketcham and Hanna 2014; Vásárhelyi et al. 2020; Withers et al. 2021).

When selecting the appropriate beam energy, the trade-off between penetration length and attenuation contrast must be considered. Low energies result in higher attenuation coefficients and higher attenuation contrasts between different phases, but also in lower penetration length, i.e., the beam is only capable of penetrating thinner samples. It must also be considered that high energies can damage the samples (e.g., because of heating). Thus, the optimal X-ray beam energy depends on the sample materials and sample size. The size of the field of view (FoV) also influences the spatial resolution, which can be quantified as the smallest distance at which the attenuation coefficient of two points is separable. The spatial resolution (R) can be expressed as:

$$R = \frac{R_d}{M} + s\left(1 - \frac{1}{M}\right) \tag{4}$$

with the resolution of the detector (R_d), the magnification (M) and the focal spot size of the X-ray source (s). The latter increases with higher X-ray tube power. The magnification can be calculated by dividing the distance between the source and detector by the distance of the source to the object. A higher resolution is achieved when the object is positioned close to the source (i.e., higher magnification), but consequently, a smaller FoV can be imaged. Therefore, the sample size determines how close the sample can be positioned to the source while still imaging the whole sample. Alternatively, only a smaller FoV can be imaged at higher resolutions (e.g., when only a portion of the sample is relevant for the experiment), or the whole sample is measured in multiple steps (in smaller FoVs), and afterwards, the datasets are stitched together. However, higher resolved datasets also need more storage space and a higher acquisition time. The spatial resolution also depends on other factors like signal-to-noise ratio or sample shape. Commonly, the incoming X-ray beam is converted into visible light by a scintillator and then converted by the charge-coupled device (CCD) detector into electrons for further digital processing (Wildenschild et al. 2002; Vásárhelyi et al. 2020; Withers et al. 2021).

The acquired 2-D projections are reconstructed into greyscale cross-sections of the object by computing the spatial distribution of the attenuation coefficient (for each voxel). This can be done by iterative or analytic methods, which are based on the Radon transform. The filtered backprojection belongs to the second group of methods and is commonly used. In this method, the 2-D projections are used to construct sinograms. Each row in a 2-D projection contains the measured beam intensities of one cross-section through the object (and each pixel holds the information along one beam path). The same rows at each projection angle are summed up in a sinogram. The sinograms are used by the backprojection reconstruction algorithm to uniformly distribute the measured intensities back at the beam ray path at each angle. However, since the intensities are distributed over the whole beam ray path, this also includes the background (i.e., empty areas). This leads to a blurred image, which is corrected with a filter (commonly the ramp filter). With a cone-beam geometry, the individual slides are not independent of each other and, therefore, depending on the angle, must be reconstructed in 3-D. The reconstruction works better when more projections are available (i.e., smaller angle changes between the projections) and the sample does not leave the FoV during rotation (Withers et al. 2021).

Higher resolution XCT scans require longer scan times (exposure time), which can be problematic when studying dynamic processes. Therefore, 4-D laboratory XCT studies are more suitable for monitoring processes that do not need a high temporal resolution. Higher temporal (and spatial) resolutions can be achieved with a synchrotron light source, which has a much higher X-ray flux and, thereby, lower exposure times are needed to achieve high resolutions (Withers et al. 2021). Other advantages of synchrotron light sources include the parallel beam geometry (which allows a more accurate image reconstruction) and the option of applying an (adjustable) monochromatic beam (in this case, no beam-hardening occurs), which enables the analysis of specific elements and the optimization of image contrast. Disadvantages of this method include limitations in beam energies, which limits the sample size, and the possibility of damaging a sample due to the high-intensity beam (Wildenschild et al. 2015; Withers et al. 2021).

2.2.2 Artefacts

Various artefacts can arise during the XCT measurement and reconstruction. These artefacts lower the image quality and make the subsequent segmentation more challenging. Depending on the extent of image artefacts, some segmentation methods (e.g., conventional global thresholding) give unacceptable results. Therefore, it is common to first apply filters to mitigate artefacts; however, these methods have to be carefully applied since they can also lead to a loss of information (Schlüter et al.

2014; Saxena et al. 2017a; Berg et al. 2018; Pot et al. 2020; Withers et al. 2021). This subsection gives an overview of common XCT artefacts and some correction strategies.

Beam hardening

Beam hardening is the most common artefact in XCT scanning and mainly causes two effects: cupping and streaking (Ketcham and Carlson 2001; Ketcham and Hanna 2014; Yadava et al. 2014).

This artefact occurs since the spectrum of a polychromatic X-ray beam changes when penetrating a sample. As the beam passes through a sample, the lower-energy components preferentially interact with the material and are more readily absorbed or scattered (in comparison to higher-energy X-rays). Consequently, the intensity of the polychromatic beam is reduced while the spectrum shifts to higher energies. The beam has a higher average energy than the original beam (and is denoted as a “harder” beam) and is then less attenuated by materials. Hence, shorter paths through a sample are proportionally more attenuating than longer paths. In cylindrical samples, the edges of the sample appear brighter (i.e., more attenuating) and the centre darker (i.e., less attenuating). This effect is illustrated in Figure 3A and can be quantified by plotting the grey values in a profile across the sample (dashed white line in Figure 3A). The result is the typical cup shape, with higher grey values at the beginning and end of the profile and lower grey values in the middle (Figure 3B). Therefore, beam hardening changes the grey value of a pixel depending on its position in the image (Ketcham and Carlson 2001; Ketcham and Hanna 2014). Furthermore, beam hardening can cause streaking artefacts (dark and bright lines) between and around dense objects in the sample. That effect can occur due to scattered radiation, photon starvation and the different extent of beam hardening, depending on the ray path (i.e., less hardening occurs when the beam only passes one dense object than when it passes two dense objects; Yadava et al. 2014; Kyle and Ketcham 2015; Park et al. 2015; Orhan et al. 2020).

Several approaches exist to correct the beam hardening artefacts during or after scanning. In principle, it would be possible to use a smaller sample or a high-energetic X-ray beam, where beam hardening could be neglected. However, the latter also results in less attenuation contrast between different materials and, therefore, difficulties in distinguishing objects (Ketcham and Carlson 2001; Wildenschild et al. 2002).

A common approach is the filtering (by installing metal filters) of low-energy X-rays before the beam hits the sample (pre-hardening of the beam). Since the whole spectrum of the X-ray beam is attenuated (in different degrees), the beam intensity is lowered, and more image noise occurs in the image (unless the scanning time is raised). The beam can also be filtered after passing the sample (post-hardening).

Another approach is the wedge calibration, where a cylindrical object with similar attenuation properties as the sample is scanned and used to calibrate the scans of the sample. (Ketcham and Carlson 2001; Scarfe and Farman 2008; Jovanović et al. 2013). Another correction method is dual-energy scanning, where the sample is scanned with different beam energies to differentiate the contributions of the photoelectric absorption and the Compton scattering to the beam attenuation (Jovanović et al. 2013; Ketcham and Hanna 2014).

The beam hardening can also be corrected during data processing. One approach is linearization, where the polychromatic attenuation data is transformed into the equivalent attenuation data a monochromatic beam would produce. Although this approach is challenging in highly heterogenous samples (Ketcham and Hanna 2014), the linearization was successfully applied in some studies (e.g., Ilzig et al. 2022). Other approaches try to adapt the reconstructed sinograms (obtained from the polychromatic beam) to sinograms that would be obtained from a monochromatic beam. For this, the travel paths of the X-ray beams have to be determined, e.g., from segmented images (e.g., Jovanović et al. 2013). Neural network approaches are also capable of removing beam hardening artefacts (e.g., Kalare et al. 2021).

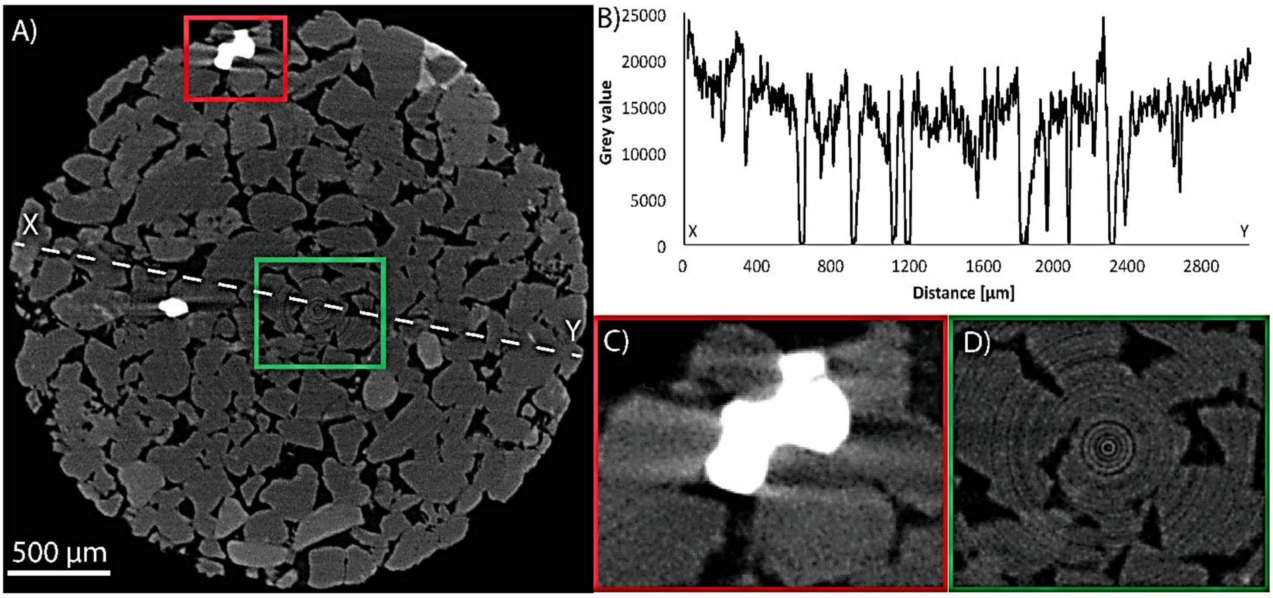


Figure 3. Common image artefacts
(A) *Exemplarily slide of a µXCT Scan of a sandstone sample, showing four common image artefacts: streaking (close-up in* **C***), ring artefacts (close-up in* **D***), noise (in both close-ups) and beam hardening.* ***(B)*** *Grey values along the profile, which are shown in (A), illustrate the cupping caused by the beam hardening artefact. The sample ("AaZ3") was measured at the JGU in the context of the HyINTEGER project. Details regarding the sample, measurements and project can be found in Hinz (2019).*

Ring artefacts

Ring artefacts are centred around the rotational axis and appear as full or partial rings (with varying radii) that are brighter or darker than their surrounding (Figure 3A, D). This artefact can be attributed to shifts in the output of detector elements. Rings are formed due to the misreading of fixed detector elements while the XCT setup moves circularly during scanning. The shift can be caused by e.g., changing experimental conditions (in respect to the calibration), the different detector response for a hardened beam (in contrast to the unattenuated beam used for the gain calibration) or a defective scintillator or detector element (Ketcham and Carlson 2001; Kyle and Ketcham 2015; Orhan et al. 2020).

Ring artefacts can be reduced during scanning by moving the detector between projections or by setting similar experimental conditions during the calibration and sample scanning (e.g., by using a filtered beam for both procedures). The wedge calibration is also suitable for reducing ring artefacts. Since ring artefacts appear as vertical lines in the sinogram, they can be removed before or after the reconstruction. However, this can also lead to the alteration of linear features in the sample (Davis and Elliott 1997; Ketcham 2006; Kyle and Ketcham 2015).

Other artefacts

Streaking artefacts (Figure 3C) can arise e.g., due to beam hardening, high attenuation by high-density objects (photon starvation) and reaching the saturation intensity of the detector (Ketcham and Carlson 2001; Withers et al. 2021). Noise (Figure 3C, D) occurs due to fluctuations in the emission and detection of photons and leads to grey value variations in homogenous areas (e.g., matrix grains). Noise can be reduced during scanning (by applying higher energies or longer exposure times), during reconstruction (with deep learning approaches) or by filtering the reconstructed images. A common noise correction filter is the non-local means filter, which eliminates noise while keeping sharp phase boundaries (Clausnitzer and Hopmans 2000; Schlüter et al. 2014; Diwakar and Kumar 2018; Guntoro et al. 2019; Orhan et al. 2020; Andrew et al. 2021).

Other artefacts include projection artefacts, phase contrast artefacts, motion artefacts or cone-beam artefacts (Davis and Elliott 2006; Cnudde and Boone 2013; Andrew et al. 2021; Withers et al. 2021).

2.3 Partial volume effect

Another complication of XCT measurements is the PVE. The grey value of each pixel in a CT image is derived from the attenuation properties of materials at the corresponding position in the actual sample. If multiple materials are present in the area represented by one pixel, the corresponding grey value of this pixel represents a mixture of the attenuation properties of these materials. This phenomenon is called the partial volume effect (PVE). The PVE appears due to the resolution limitations of XCT measurements (Figure 4) and leads to a blurring of material boundaries (Ketcham and Carlson 2001).

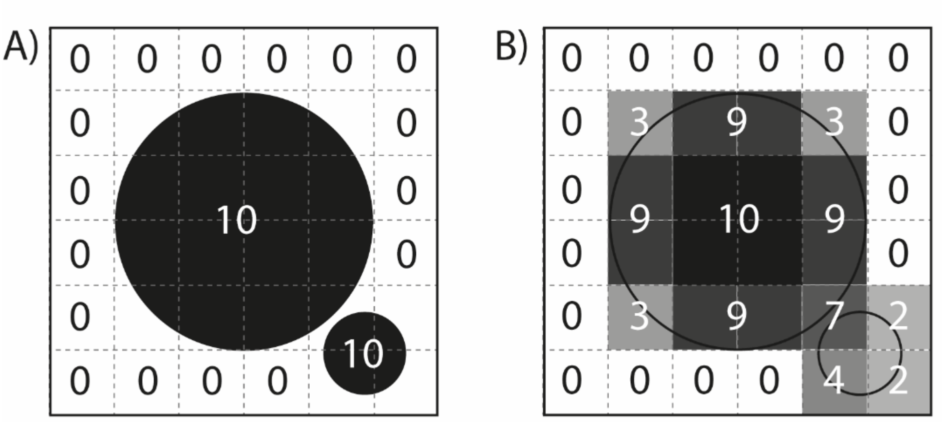


Figure 4. Illustration of the partial volume effect
***(A**) Two circular objects are to be measured. **(B)** Due to the finite pixel resolution, the resulting grey values in the measured image result either only from the background (0), only from the objects (10), or are a mixture of both grey values (between 0 and 10). The pixel grid is overlain in both images. Adapted after* Soret et al. (2007).

The PVE gains influence in an image with low (i.e., coarser) spatial resolutions, small structures and near material boundaries (Ketcham et al. 2010; Garfi et al. 2020; Yang et al. 2020). Thus, in the case of fractures, the PVE has the greatest influence on low aperture areas and surface details (roughness). This effect can complicate the segmentation process (e.g., Garfi et al. 2020) and result in an over- or underestimation of fracture aperture, but it can also be exploited to gain additional information (Johns et al. 1993). The PVE can be corrected if additional data is available (e.g., Kato et al. 2013; Ramandi et al. 2017; Yang et al. 2020). Recent advances in deep learning techniques also include the application of the PVE correction in the pore space (Wang et al. 2022). Although machine learning has great potential, there are still some limitations that must be overcome (discussed in subsection 2.4.2). In the case of fracture detection, there exist four popular methods to correct for the PVE and extract information about the true fracture volume:

(i) The missing attenuation (MA) method is based on the attenuation characteristics of the present materials, i.e., the X-rays are less attenuated by the fracture volume compared to the matrix volume. Consequently, CT numbers (or grey values) of pixels can be used to distinguish between pixels influenced by the presence of the fracture and the degree of the influence (i.e., the local fracture aperture). The MA method is suitable for small apertures (with respect to the image resolution), i.e., with highly blurred fracture areas, since CT numbers of all pixels affected by the presence of the fracture are integrated to determine the local fracture aperture. Furthermore, the application to heterogenous rocks is possible, and a calibration is not necessary (Ketcham et al. 2010; Huo et al. 2016).

(ii) The peak height (PH) method utilizes the negative peak in CT numbers caused by the lower attenuation of the fracture volume compared to the surrounding matrix (e.g., employed in Watanabe et al. 2011). This method requires careful calibration (e.g., by measuring CT values of gaps with known width) and is usually applicable in homogenous materials and for small fractures (Ketcham et al. 2010).

(iii) The full-width-half-maximum method (FWHM) uses the midpoint between the CT numbers of the rock matrix and air to calculate the aperture. This approach is most suitable for apertures larger than the spatial resolution (Ketcham et al. 2010) and less accurate compared to the MA method (Van Geet and Swennen 2001). Cappuccio et al. (2020) successfully combined the FWHM with the MA to extract a fracture network with mean apertures slightly above the image resolution. They calibrated the MA with information from the FWHM analysis in fracture apertures above the spatial resolution, to calculate the sub-resolution apertures in the fractures.

(iv) The inverse-point-spread function (IPSF) method is a mixture of the MA and PH methods and employs the IPSF to calculate the apertures of homogenous and heterogenous materials. Studies showed promising results from this method; however, it is computationally expensive and needs a calibration (Ketcham et al. 2010; Mostaghimi et al. 2017; Zhou et al. 2021).

The main disadvantage of most methods is the need for a calibration, which can be done with e.g., SEM images (Ramandi et al. 2017), spacers (Kling et al. 2016) or fractures with a unique and known aperture (Watanabe et al. 2011). Additionally, most methods require resolved fracture apertures (i.e., the spatial resolution must be smaller than the apertures). The CFMA method allows the calculation of sub-resolution apertures and was successfully applied in various studies (Huo et al. 2016; Kling et al. 2016; Wenning et al. 2019, 2021; Shen et al. 2021). However, uncertainties of about 20 – 60 µm were reported (Huo et al. 2016; Wenning et al. 2019) and Kling et al. (2016) demonstrated that even small deviations in the CT value of the matrix can have a significant influence on the permeability.

Kling et al. (2016) used a simplified MA method with calibration data to reconstruct a 3-D fracture. Shen et al. (2021) applied the CFMA method to calculate fracture volumes. Ramandi et al. (2017) created a 3-D fracture network reconstruction in higher resolution than provided by the µXCT-images.

However, they applied a calibration method based on SEM images. So far, no study has applied the CFMA method without the use of calibration data (from additional measurements) to reconstruct a single 3-D fracture in higher resolution.

2.4 Segmentation

Segmentation refers to the identification and labelling of distinct phases (like pores or minerals) in grey value images. Segmentation is a crucial step in the DRP workflow, affecting the qualitative and quantitative analysis of digitized samples (Iassonov et al. 2009; Andrä et al. 2013a, b; Leu et al. 2014; Saxena et al. 2017a; Pot et al. 2020). An ideal grey value histogram would show distinct peaks for each phase (i.e., a multi-modal distribution). Due to imaging artefacts or similar phases, histograms are usually complex with continuous distributions or overlapping peaks, which makes proper segmentation challenging (Al-Marzouqi 2018; Berg et al. 2018). The quality of the segmentation is usually quantified by comparing simulated rock properties with experimental data (e.g., porosity, permeability, capillary pressure curves, pore size distribution). In order to ensure a meaningful validation, more than one parameter should be considered (Leu et al. 2014). The segmentation result is usually not unique (i.e., more than one result shows similar properties as the real sample), which in turn introduces uncertainties in calculations based on the segmentation result (Krygier et al. 2021).

In order to avoid errors caused by the segmentation, alternative methods were proposed to directly compute properties from the CT numbers of grey value images. Taud et al. (2005) developed an algorithm to compute porosity from the distribution of CT numbers (or grey values). Goldfarb et al. (2017; 2022) assigned voxel densities according to CT numbers by applying a calibration curve based on measurements of materials with a known density. By assuming a linear relationship between density and porosity, a voxel porosity was assigned. Based on this, the Effective Medium Theory was applied to calculate shear and bulk modulus for each voxel. Furthermore, the wave propagation through the sample was simulated in their studies. Their approach was validated with laboratory measurements, and their results demonstrate that the segmentation-less methods can give better results than workflows using segmented images (Goldfarb et al. 2017; 2022). Ikeda et al. (2020) calculated physical rock properties without scanning reference materials. These approaches are computationally efficient (especially for large datasets), suffer less from user bias, and do not need ultra-high spatial resolution (Goldfarb et al. 2022). However, only a limited number of properties can be calculated, and the approach was not compared with more advanced segmentation methods (i.e., machine learning approaches).

Other segmentation-less workflows include the analysis of dynamic features by directly using the measured sinograms (Bührer et al. 2020) or the automatic detection of fracture skeletons (Liang 2016) or fractures in the grey value images (e.g. Ramandi et al. 2022). Although automatic fracture detection seems promising, the approach was not compared with advanced segmentation methods, and it remains unclear if the obtained fracture apertures are realistic (Ramandi et al. 2022).

Due to these shortcomings, segmentation is still a frequently applied step in the DRP workflow. Many methods exist and can be categorized as conventional (Iassonov et al. 2009) or machine learning approaches (Chauhan et al. 2016), which also include recently developed methods that are based on CNNs (e.g., Wang et al. 2021b). Numerous studies concluded that machine learning approaches have the potential to overcome the shortcomings of conventional methods and provide superior segmentation results (e.g., Andrew 2018; Berg et al. 2018; Karimpouli and Tahmasebi 2019; Varfolomeev et al. 2019; Wang et al. 2021b).

2.4.1 Conventional methods

Conventional segmentation methods can be based on local or global differences in grey values. An extensive overview of common methods is given by Iassonov et al. (2009) and Schlüter et al. (2014). Global segmentation methods are based on the global analysis of image features like the distribution of grey values or the distribution of grey value gradients. By setting thresholds in the distribution, voxels are categorized in defined phases. Thresholds can be determined manually or by applying algorithms that find optimal threshold values. These algorithms include the Otsu method (Otsu 1979), K-means clustering (MacQueen 1967), MaxEntropy (Kapur et al. 1985) or the Liu-Song algorithm (Song et al. 2020). Local methods consider spatial information of the neighbouring voxels and are applied locally on images. These methods include the watershed method (Beucher and Meyer 1993) or workflows where the indicator kriging is applied (Oh and Lindquist 1999).

Global thresholding methods struggle with overlapping features, which are totally or partially falling within the same range of grey values (Berg et al., 2018). Local methods are considered more accurate and robust against changing input parameters and noise; however, local and global methods are both often time-consuming, susceptible to operator bias and have problems with handling image artefacts or complex grey value distributions. Furthermore, the segmentation quality can be significantly impacted by the choice of the segmentation method and variations in input parameters (Taud et al. 2005; Iassonov et al. 2009; Leu et al. 2014; Schlüter et al. 2014; Saxena et al. 2017a; Varfolomeev et al. 2019, Shou et al. 2020; Reinhardt et al. 2022). The application of image filters can improve the performance of conventional segmentation methods (e.g., by noise reduction before the segmentation); however, filtering can also lead to a loss of information (Saxena et al. 2017b; Berg et al. 2018; Pot et al. 2020).

2.4.2 Machine learning methods

Neural Networks

The application of machine learning methods in the DRP workflow quickly advanced in recent years (Rabbani et al. 2021). In addition to well-established machine learning methods (e.g., random forest (RF)), neural networks were adapted for many tasks in porous material research, including lithological classification (Baraboshkin et al. 2020); rapid prediction of material properties (e.g., permeability, tortuosity or the flow velocity field) from 3-D images (e.g., Rabbani et al. 2020; Santos et al. 2020); the generation of synthetic samples, reconstruction and artefact removal (e.g., Mosser et al. 2017; Feng et al. 2019; Shams et al. 2020; Andrew et al. 2021; Tawfik et al. 2022; Zheng and Zhang 2022; Li et al. 2022); enhancing resolution (e.g., Wang et al. 2018; Wang et al. 2020a; Ahuja et al. 2022; Alqahtani et al. 2022); and two or multi-phase segmentation (e.g., Karimpouli and Tahmasebi 2019; Karimpouli et al. 2019; Varfolomeev et al. 2019; Phan et al. 2021; Tang et al. 2022; Lee et al. 2022).

Neural networks are composed of an input layer, output layer and various hidden layers in between. The layers consist of many computational units called neurons, which receive input from the neurons of the previous layer and output a value to the next layer. The neural network estimates the relationship between inputs and outputs by assigning weights and a bias to the inputs of each neuron. Nonlinearity is achieved by introducing activation functions, like the rectified linear unit (ReLU) or the sigmoid function. The weights and biases are iteratively updated and optimized, using the gradient-descent method and a cost function to quantify the prediction error (Goodfellow et al. 2016; Feng et al. 2019; Rabbani et al. 2021). Deep learning methods are characterized by their multiple layers of nonlinear information processing and their feature representation at successively deeper and more abstract layers. The advantage of a deeper architecture is to extract and use more abstract information (Deng and Yu 2014; Guo et al. 2016; Karimpouli and Tahmasebi 2019). CNNs belong to the deep learning methods and are characterized by convolutional layers, which extract features by applying the convolution operator (Badrinarayanan et al. 2017).

Special types of neural networks are encoder-decoder networks, where the encoder part subsequently extracts more abstract image features (by downsampling the feature map), and the decoder part uses the features to make a prediction (by upsampling). Encoder-decoder networks in the form of CNNs were successfully applied in the segmentation step of the DRP workflow (e.g., Karimpouli and Tahmasebi 2019; Varfolomeev et al. 2019; Byun et al. 2021). Lee et al. (2022) evaluated two machine-learning-based methods (U-net and RF) and three conventional methods for fracture segmentation. Among them, the 2-D U-net model provided the best performance regarding the segmentation quality and processing time. Studies indicated that a 3-D CNN is more robust and accurate than a 2-D CNN since the former extracts the features in a 3-D space (Karimpouli et al. 2019). Furthermore, increasing the complexity of CNN architectures has positive effects on their performance (Karimpouli and Tahmasebi 2019).

Generating a ground truth for the training procedure remains challenging. The segmentation results from conventional methods can be used as ground truth (Niu et al. 2020; Lee et al. 2022). However, as discussed above, these segmentations depend on the applied method, suffer from user bias and are less reliable in the presence of artefacts. Therefore, flaws in the provided ground truth can influence the quality of the CNN segmentation (Niu et al. 2020). Karimpouli et al. (2019) and Wang et al. (2021b) prepared the ground truth manually, which is time-consuming and also prone to user bias. Furthermore, the quantity and quality of the training data are crucial for the quality of the CNN segmentation. For example, data augmentation or the generation of synthetic data can be applied to increase the quantity of available training data (Karimpouli and Tahmasebi 2019; Wang et al. 2021a). Without prior filtering, CNNs provide good segmentation results (and avoid user bias) if appropriate training data is available. However, it can be challenging to get sufficient training data of good quality. This is especially troubling with unique samples (Hiramatsu et al. 2018; Karimpouli et al. 2019; Varfolomeev et al. 2019; Niu et al. 2020). Besides the availability of sufficient training data, other main challenges of deep learning approaches include the lack of standard benchmark datasets, high computational demands, the standardization of performance measures, accuracy limits, and their extrapolative potential (Wang et al. 2021a).

Random Forest

Interactive machine learning algorithms, using classifiers like the RF, were also successfully applied for segmenting geological data (e.g., Berg et al. 2018; Jacob et al. 2021; Reinhardt et al. 2022). If the classifier is appropriately trained, it can handle image artefacts (i.e., no prior filtering step is mandatory), and it can perform better than conventional methods (Berg et al. 2018; Scanziani et al. 2018; Kovács et al. 2019; Garfi et al. 2020).

The training data for each phase is usually provided by the user via brush strokes, directly on images of the dataset, or a similar one. Therefore, the segmentation does not require extensive, additional training data. In order to learn the characteristics of each segmented phase, several image filters with varying parameters are applied. This allows the extraction of features from the images. These features provide additional (i.e., besides the grey value information) and more abstract information about each phase and allow the classifier to better distinguish the phases (Arganda-Carreras et al. 2017; Berg et al. 2019).

A filter is applied to an image by convoluting the image with a smaller matrix called kernel (e.g., Gupta and Chandel 2013). Figure 5 presents an overview of commonly applied filters in interactive machine learning methods. The resulting images, after applying the corresponding filter on the grey value image (Figure 5A), are shown. The Gaussian smoothing filter (Figure 5B) uses a kernel based on the Gaussian distribution and is applied to blur images, thereby removing noise and details (Hsiao et al. 2007). The Laplacian of Gaussian filter (Figure 5D) combines the Gaussian smoothing with the Laplacian filter, where the former filter smooths the image, and the latter enhances edges. The prior smoothing step is necessary because the Laplacian filter is error-prone in the presence of noise. The applied kernel is based on the Laplacian operator, which is applied to the Gaussian distribution. This filter is used for edge and blob detection (Gonzales and Woods 2007; Kong et al. 2013). The difference of Gaussians filter (Figure 5C) is a computationally efficient approximation of the Laplace of Gaussians filter and is therefore also used for edge or blob detection. The kernel is based on the difference of two Gaussian distributions (Wang et al. 2012; Kong et al. 2013). The (2-D) Gaussian gradient magnitude filter (Figure 5E) is used for edge detection. Its kernel is based on the first derivatives of the Gaussian distribution in the X- and Y-directions (Sanders et al. 2016; Kurale and Vaidya 2017). The Hessian of Gaussian eigenvalues filter (Figure 5F, G) is based on the second-order partial derivatives of a Gaussian distribution, which are represented in the Hessian matrix and convoluted with the image. Eigenvectors of the Hessian matrix describe the direction of curvature, and corresponding eigenvalues describe their magnitude. In 2-D, one eigenvalue describes the largest contrast change (locally), and the second eigenvalue represents the contrast change orthogonal to the first one. The information of both eigenvalues can be used to gain structural information (e.g., tubular structures in the image). Applying this filter can give two sets of images, one with the bigger eigenvalue and one with the smaller eigenvalue (Voorn et al. 2013; Mao et al. 2021). The Structure tensor is a matrix composed of the gradient (first-order partial derivatives) of a Gaussian distribution, convoluted with the image. The two corresponding eigenvalues (each gives one set of images) are used for the structure tensor eigenvalues filter (Figure 5H, I) and can be utilized to detect edges and corners (Baghaie and Yu 2015). The strength of all these filters can be changed by setting the standard deviation (σ) of the Gaussian distribution.

The information from the grey value and filtered image are used by a classifier to create a segmentation (Figure 5J).

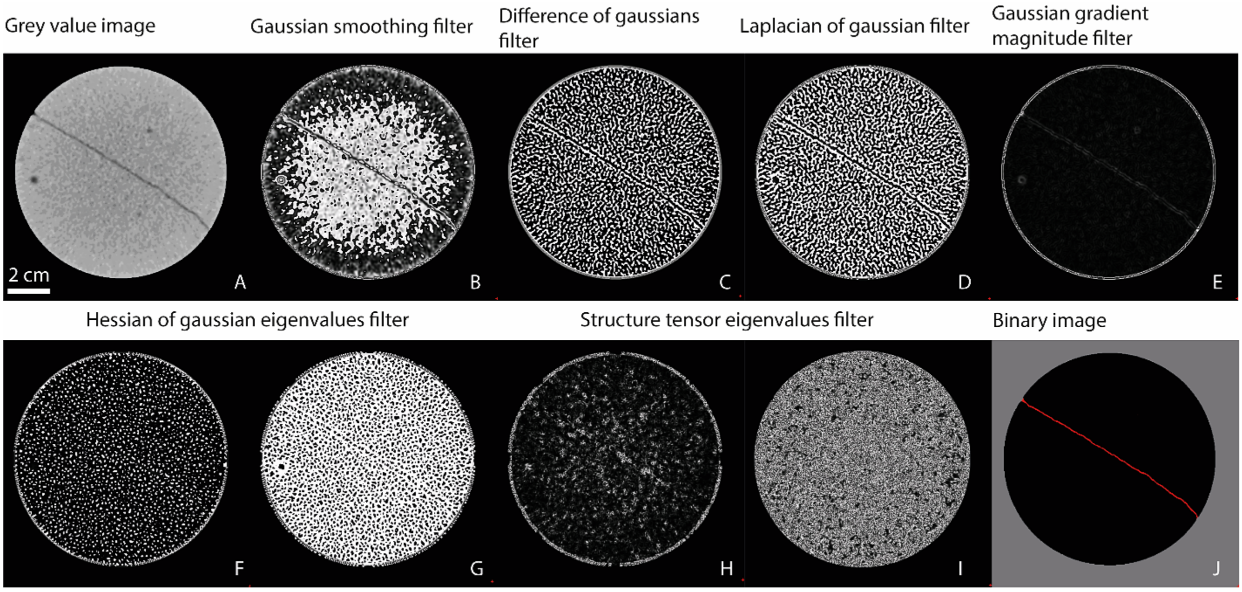

Grey value image
Gaussian smoothing filter
Difference of gaussians filter
Laplacian of gaussian filter
Gaussian gradient magnitude filter
2 cm
A
B
C
D
E
Hessian of gaussian eigenvalues filter
Structure tensor eigenvalues filter
Binary image
F
G
H
I
J

Figure 5. Overview of commonly applied filters in the interactive machine learning workflow
*Filters are applied to the grey value image **(A)** to obtain features that allow for a finer distinction between the present phases. The Gaussian smoothing filter **(B)**, difference of Gaussians **(C)**, Laplacian of Gaussian **(D)**, Gaussian gradient magnitude **(E)**, Hessian of Gaussian eigenvalues **(F, G)**, and structure tensor eigenvalue **(H, I)** are applied with a sigma value of 1 on the grey value image (A) to obtain the features used to generate the segmentation result **(J)**. The filters in the Ilastik software are implemented as described in https://ukoethe.github.io/vigra/doc-release/vigra/group__ConvolutionFilters.html#gab80e356fa487f97d718ed815a3cf4855.*

Filters are applied to the grey value images to obtain features, which are used by an RF classifier to create a segmentation. The classifier is composed of multiple, uncorrelated decision trees, which are built by using bagging (bootstrap + aggregating) and feature randomness (Breiman 2001). Each decision tree splits the data at decision nodes (using a random subset of features) until a leaf node is reached, where a class is assigned. The threshold for splitting the data can be chosen randomly or optimized by considering metrices like entropy or Information Gain. For each decision tree, a random subset of the training data (provided by the user via brush strokes) is selected by bootstrapping. Each decision tree makes a prediction for each voxel, and the final class assignment is decided by the majority of the votes (aggregating). Since the individual trees are trained on a random subset of the training data, there exists unused training data for each tree (out-of-bag dataset). This dataset is used to quantify the quality of the RF by evaluating the predictions of the RF on this dataset (since it is training data, the correct result is known). The ratio of incorrectly classified out-of-bag data is called out-of-bag error. This measure allows the comparison of different RFs and thus optimization of the applied RF without the need for an additional validation dataset (Biau and Scornet 2016).

Popular implementations of the interactive RF classifier include the Trainable Weka Segmentation (TWS; Arganda-Carreras et al. 2017) and the software package Ilastik (Berg et al. 2019); both are open-source. Compared to the Ilastik implementation, the TWS has higher computational demands and requires a deeper knowledge of the applied methods since the user must decide on the classifier and filters used. In contrast, Ilastik offers a predefined workflow (Berg et al. 2019).

2.5 Fluid dynamics

This section presents the fundamental laws to describe fluid dynamics assuming Newtonian fluids, i.e., the fluid viscosity (the shear stress is linearly related to shear rate, Batchelor 2000) and density (i.e., incompressible fluid) are constant. The physics of the fluid flow at the pore scale obeys the laws of momentum and mass balances. The mass balance is expressed as $\nabla\vec{u} = 0$, with the 3-D fluid flow velocity field $\vec{u}$. In order to conserve mass, the change of flow velocity in one direction (X, Y, or Y) must be compensated by the change of flow velocity in the other two directions.

The momentum conservation is described with the Navier-Stokes equation:

$$-\mu\Delta\vec{u} + (\rho\vec{u} * \nabla)\vec{u} + \nabla p = \vec{f} \tag{5}$$

with the dynamic fluid viscosity (μ), the fluid density (ρ), the pressure (p), and the force density field ($\vec{f}$). The Stokes part of the equation considers the viscous force, while the Navier part corresponds to the inertial effects (i.e., convective acceleration). Due to small changes in fluid composition at the pore scale, the density is assumed to be constant, and effects due to gravity are neglected (Noiriel and Soulaine 2021). For slow flow rates (usually applicable for flow in aquifers and reservoirs), the Navier-Stokes equation can be further simplified by neglecting the inertial effects (pressure drop is linearly related to the mean velocity). The result is the Stokes equation (Blunt 2017):

$$-\mu\Delta\vec{u} + \nabla p = \vec{f} \tag{6}$$

The Reynolds number (Re) is the ratio between inertial and viscous forces and is given by:

$$Re = \frac{\rho v l}{\mu} \tag{7}$$

with the typical flow speed (v) and the characteristic length scale (l), which can be a representative pore length in the context of porous media or the mean aperture in flow through fractures (Blunt 2017; Stoll et al. 2019). Equations 5 and 6 are recommended for fluid flows with a low Reynolds number. Darcy's law can be applied at the laminar flow condition to calculate the permeability (K) by using the calculated force density field (Darcy 1856):

$$\vec{u} = -\frac{K}{\mu}(\nabla p - \vec{f}) \tag{8}$$

The permeability is a property of the medium and expresses the potential of fluids or gases to flow through the porous/fractured medium (Schwartz et al. 1993). The fraction of void space in a porous medium is expressed as porosity. Fluid flow through such a medium is only possible if the pore space is connected (Hölting and Coldewey 2005). Another important parameter in this context is the tortuosity, which describes the deviation of the actual flow path length of the straight distance between two points (e.g., Clennell 1997; Xiao et al. 2013). Alternative approaches to deviate properties of fractures (e.g., permeability, connectivity) include the persistent homology analysis, where topological and geometrical information is used for these estimations (e.g., Suzuki et al. 2021).

For solving the Navier-Stokes and Stokes equation, boundary conditions (BCs) are applied to the computational domain. BCs reduce unwanted boundary effects by defining the fluid behaviour when it encounters an obstacle or a boundary of the computational domain. Common BCs include the no-slip condition on the surfaces of solids ($\vec{u} = 0$) or the symmetric and periodic BCs, which extend the computational domain for the calculations (Hilden et al. 2020). Several computational fluid dynamics software (e.g., OpenFOAM (OpenCFD Ltd), GeoDict (Math2Market GmbH) or COMSOL Multiphysics (Comsol Multiphysics GmbH)) exist to compute fluid flow in 3-D geometries.

2.6 Particle transport

Suspended solid phases are traditionally classified based on their size; however, multiple definitions for the size ranges exist in the literature (Lead and Wilkinson 2007). Commonly, solids with a diameter between 1 nm and 1 µm (up to 10 µm) are defined as colloids. Additionally, the term nanoparticle can be applied to colloids with a diameter below 100 nm. Due to the different shapes of solid phases, an equivalent diameter can be applied to compare different solids (Lead and Wilkinson 2007; Zhang et al. 2012; Molnar et al. 2019). The definition of dispersed solid phases can also be based on transport behaviour. In contrast to molecules, the diffusion of colloids can be expressed by Brownian motion (due to collisions, expressed as a random walk), the forces of interaction can be bigger, and their motion is more determined by the flow (i.e., colloids follow the flow trajectories more strictly than molecules). For bigger solids (commonly referred to as particles), the influence of diffusion diminishes, and the specific surface area (which influences the reactivity) gets smaller (Molnar et al. 2019). Sources of suspended particles include precipitation, physical and chemical erosion, or flushing of material that coats minerals or (partially) fills fractures/pores (Molnar et al. 2019). The investigation of suspended solid materials is motivated by e.g., their clogging potential of porous/fractured media or their potential to transport contaminants (Ryan and Elimelech 1996; Cumbie and McKay 1999; Valsala and Govindarajan 2022).

In pore-scale modelling, the transport of particles in fluids is characterized by advection and diffusion (e.g., Bijeljic et al. 2004; Rolle et al. 2012). The change of concentration (c) over time (t) is expressed by the advection-diffusion equation, which is based on a Fickian law (e.g., Bauget and Fourar 2008; Li et al. 2008b):

$$\frac{\partial c}{\partial t} = \nabla(D\nabla c) - \nabla(\vec{u}c) \tag{9}$$

with the molecular diffusion coefficient (D). The particle movement due to advection depends on the fluid velocity, whereas the Brownian diffusion depends on the particle concentration and the diffusion coefficient. With smaller particle sizes, the influence of Brownian motion gets more important (Molnar et al. 2015). Usually, in fractured or porous media, advection dominates along the flow paths (e.g., in a fracture), whereas diffusion dominates in porous domains that do not support the main flow paths (Noiriel et al. 2007). The advection transports particles along streamlines, whereas diffusion allows particles to deviate from them (Molnar et al. 2019).

Usually, during transport, a spatial spreading (dispersion) occurs. In single fractures, the dispersion can be attributed to (i) the parabolic fluid velocity profile between the fracture surfaces (Taylor-Aris dispersion), which describes the dependence of flow velocity on the distance to the solid (i.e., velocity is zero at the boundary, and highest in the middle of the fracture). (ii) Roughness and (iii) aperture variability (and channelling) further modulates the velocity distribution in the fracture (i.e., leading to more heterogenous flow fields with nonlinear features like eddies). Furthermore, (iv) Brownian diffusion, (v) matrix diffusion, or interactions with the rock due to (vi) sorption can influence the dispersion (e.g., Roux et al. 1998; Bodin et al. 2003b, a; Boutt et al. 2006; Blunt 2017; Briggs et al. 2017; Zou et al. 2017; Stoll et al. 2019). The dispersion can be analysed with the characteristics of a breakthrough curve, where the particle concentration is plotted against the time. These features include breakthrough time, number of peaks, peak heights, shape, or tailing behaviour (e.g., Bodin et al. 2003b; Chen et al. 2017; Stoll et al. 2019; Frank et al. 2020b). The conventional advection-diffusion equation fails to predict the transport behaviour of heterogeneous media because of the occurrence of non-Fickian behaviour, including an earlier breakthrough, longer tails, and multiple peaks (e.g., Chen et al. 2017; Dou et al. 2019; Wang et al. 2020b). Several studies demonstrated that the continuous-time random walk theory is capable of overcoming the shortcomings of the classical advection-diffusion equation (e.g., Bauget and Fourar 2008; Berkowitz and Scher 2009; Wang and Cardenas 2015;

Dou et al. 2019; Frank et al. 2020a). At long resident times or length scales large compared to the size of heterogeneities, the non-Fickian behaviour transitions into Fickian behaviour, nevertheless, usually, these scales are large, and therefore the non-Fickian behaviour must be considered (e.g., Berkowitz and Scher 2009).

In this study, the AddiDict module of the GeoDict 2020 software package was applied to compute the movement of molecular (tracer particles) and finite-sized particles (barite particles). The equations presented below are shown as they are implemented in the simulation software. The capabilities of the GeoDict software were demonstrated by various studies (e.g., Chagneau et al. 2015; Glatt et al. 2015; Khan 2014; Lenchenkov et al. 2018; Hinz 2019).

The transport of molecular particles is described by the advection (based on the previously computed flow field) and (Brownian) diffusion. However, additional influences on particle transport could be considered and include electrostatic forces, external forces, diffusion into the rock matrix, and particle-particle interactions. The advection is the bulk movement of the liquid, i.e.; the particles are transported along with the streamlines of the fluid flow (Clennell 1997). The Brownian motion describes random particle movement (i.e., deviation from the streamlines) due to collisions within the fluid (Njuguna et al. 2021). The diffusive motion can be expressed with equation 13 and the Stokes-Einstein relation (equation 10; Einstein 1956):

$$\Delta x_p = \sqrt{2D * t} \qquad (10)$$

where x_p is the travelled distance of a particle.

The molecular particle motion ($\vec{v}$) can be obtained by adding the flow velocity ($\vec{u}$, therefore accounting for advective transport) with the random Brownian motion (accounting for diffusion) for each time step (dt). The Brownian motion is simulated by applying a random walker combining a probability function (the 3-D Wiener measure $\overrightarrow{W}(t)$, which is used to mimic Brownian motion in 3-D) and equation 10 (Rief et al. 2006; Spivakovskaya et al. 2007; Constantin and Iyer 2008; Weber et al. 2020):

$$\vec{v} = \vec{u} + \sqrt{2D}\frac{d\overrightarrow{W}(t)}{dt} \qquad (11)$$

The transport of particles with a defined size (i.e., not molecules) is computed by the following equations 12 - 14, which consider the mass (m) and the radius (r) of the particles (Weber et al. 2020):

$$m\frac{d\vec{v}}{dt} = \gamma\left(\vec{u} - \vec{v} + \sqrt{2D}\frac{d\vec{w}(t)}{dt}\right) \qquad (12)$$

$$D = \frac{k_B T}{\gamma} \tag{13}$$

$$\gamma = 6\pi\mu \frac{r}{C_c} \tag{14}$$

with Boltzmann constant (k_B), the temperature (T), the friction coefficient (γ), and the Cunningham correction (C_c), which corrects for drag forces acting on very small particles that move through a fluid (Cunningham 1910). The Cunningham correction is usually not necessary for particle transport in water (Weber et al. 2020). Note: the presence of particles does not influence the velocity field in this approach.

For the interaction of particles with solids, three collision models can be considered. In the caught-on-first-touch model, particles are simply deposited when they collide with a solid. In the sieving model, the particles do not stick to the solid surfaces but can be caught if they do not move anymore or are in contact with the solid at two different points. The restitution parameter is used to determine the energy loss at a collision in the following manner: a restitution value of 0.3 would result in an energy loss of 70 % at a collision, and the particle is reflected with a 70 % lower speed than it had before the collision. Restitution values of 0 and 1 would result in a complete or no energy loss at a collision, respectively. The Hamaker model considers adhesive forces, i.e., particles stick to the surface if their velocity is sufficiently small. The criterion is given by:

$$v^2 < \frac{H}{4\pi\rho a_0 r^2} \tag{15}$$

with the adhesion expressed as the Hamaker constant (H) and the adhesion distance (a_0, in AddiDict a constant value of 4×10^{-10} m is implemented). Since the particle radius is considered in this collision model, it is only available for finite-sized particles. The Hamaker model also considers the energy loss of a particle when colliding (i.e., restitution) with a solid phase (Hamaker 1937; Rief et al. 2006; Weber et al. 2020).

The dimensionless Péclet number (Pe) is a common measure to describe the dominant transport mechanism in a system. Pe is defined as the ratio of advection and diffusion:

$$Pe = \frac{u * l}{D} \tag{16}$$

Hence, diffusion is the dominant transport mechanism at Pe < 1, and advection dominates at Pe > 1 (e.g., Stoll et al. 2019).

BCs of simulating particle transport include the reflective BC (particles are reflected at this boundary), the periodic BC (particles leave at this boundary and enter the domain at the opposite side), and the open BC (particles are allowed to leave the model domain at this boundary). Particles leaving the computational domain at the latter boundary are counted in the computed breakthrough curve.

3. Materials & Methods

3.1 Samples

Two different sandstone samples were obtained within the framework of the ReSalt project and used in this subsequent study. A Flechtinger sandstone was chosen as an analogue material of the geothermal reservoir rock at Groß Schönebeck (Heiland 2003; Blöcher et al. 2014, 2016). The Flechtinger samples originated from a quarry in Bebertal (Saxony-Anhalt, Germany) at the Flechtinger mountain ridge. In previous studies, it was described as a well-consolidated, well-sorted, cross-bedded, and fine-grained Permian sandstone with a reddish-brown colour (Heiland 2003; Hassanzadegan et al. 2012). The porosity was determined in a range from 6 – 11 %, with mean grain sizes between 0.1 – 0.4 mm, and the matrix permeability ranging from 10^{-17} – 10^{-16} m² (Heiland 2003; Blöcher et al. 2009; Stanchits et al. 2011; Hassanzadegan et al. 2012). The second sandstone type used in this study is the Remlinger sandstone, forming a Triassic formation at the Thüngersheimer Anticline. The samples were taken from a quarry in Remlingen, near Würzburg (Bavaria, Germany). The mean porosity of the drilled sample cores was found to be slightly higher than the Flechtinger samples (approximately 12.9 %), whereas a similar mean matrix permeability of 5.50×10^{-17} m² was determined by Frank et al. (2020b). Visually, the Remlinger sandstone was described as more homogeneous and isotropic than the Flechtinger sandstone (Frank et al. 2020b). Due to the low matrix permeability (i.e., water flow through the matrix could be neglected), like the Flechtinger samples, this sandstone was suitable for analysing the fluid flow and the hydro-mechanical particle deposition in fractures.

For both sandstone types, five samples with dimensions of about 9.9 x 15 cm (diameter x length) were prepared for the subsequent DRP investigation. Samples were drilled on different blocks from each quarry, with a drilling direction perpendicular to the stratification. Single tensile fractures were generated by applying a Brazilian test (e.g., Wang et al. 2020c) over the length of the samples. After splitting, both core halves of each sample were covered with a shrinking tube (PTFE 400-101, 6/25, 4mm (4'')) to stabilize the cores and avoid unwanted displacement during the transportation and experiments. Hereinafter, the samples are named in the following way: the first three letters indicate if the sample is a Flechtinger ("Fle") or a Remlinger ("Rem") sandstone, the first digit represents the number of the quarry block, followed by the number of the drilled sample from that corresponding block. For example: "Fle_8_S2" labels the second Flechtinger sandstone core drilled from block number 8.

3.2 Computed Tomography

Ten fractured sandstone cores were scanned with the medical XCT instrument “SOMATOM Definition AS VA48A instrument”, located at the Montanuniversität in Leoben. As beam settings, an energy of 120 keV and 90 mA was applied, and for the reconstruction, the associated device software (Somaris/7 syngo CT VA48A) was used. The reconstructed image stacks exhibit grey values normalized to the Hounsfield scale, which is commonly used in medical CT fields and designed to emphasise the contrast between air, water, tissue, and bone (Hounsfield 1980). This scale typically ranges from -1000 (typical for air) to over 2000 (typical for bones) and is expressed in Hounsfield Units (HU, Zatz 1981). The HU of a material is calculated by multiplying 100 by the difference in the linear attenuation coefficients of the material and water, divided by the difference in linear attenuation coefficients of water and air (Vásárhelyi et al. 2020). Before the segmentation, these images were pre-processed in the following steps, using the 3-D visualization software Avizo 2019.3 (FEI Visualization Sciences Group, France): (i) the Lanczos filter was applied to resample the images to a uniform voxel size of 200 µm (Lanczos 1950), (ii) the image type was converted to 16-bit (unsigned), and (iii) the grey values were raised by 2000 to eliminate negative grey values. The resulting image stacks had dimensions of 570 x 570 x 789 voxels. For the segmentation method comparison (sections 3.3 and 4.1), the pre-processed images were additionally masked, i.e., the background voxels were set to a grey level value of zero.

3.3 Segmentation

3.3.1 Conventional methods

In order to further analyse the dataset and run simulations, the grey value image stacks must be converted into binary images. Since this step is crucial for the following analysis and can introduce significant errors, the segmentation method must be chosen carefully, and the results must be adequately validated (Leu et al. 2014). That is particularly the case with grey value histograms that show a continuous distribution without distinct peaks for the occurring phases. Since that behaviour was observed for each sample (Figure 6), the segmentation was expected to be challenging, especially for conventional methods (Al-Marzouqi 2018). Therefore, for each sandstone type, one representative sample was selected (Fle_8_S1 and Rem_1_S2) to evaluate the performance of conventional segmentation methods (i.e., global thresholding techniques, watershed segmentation) against more advanced machine learning approaches (RF classifier and CNN).

For both samples, global thresholds were computed by using the Otsu method (Otsu 1979) and K-means clustering (MacQueen 1967), as implemented in GeoDict 2020 (Math2Market GmbH, Kaiserslautern, Germany). The Otsu method is a histogram-based thresholding technique which determines a global threshold by minimizing the in-class variance. K-means clustering identifies a number of centroids (given by the user) and clusters the available data points around their nearest centroid. Iteratively, the position of the centroids is updated to minimize the squared Euclidean distances. Additionally, several manual segmentations were created by manually choosing different suitable global thresholds. For each segmentation, the minimum grey value was set to 10. For the Rem_1_S2 sample, maximum grey values of 3350, 3380, 3400, 3410, 3420, 3430, 3440, 3582 (Otsu method), and 3588 (K-means clustering) were used (Figure 6D). For the Fle_8_S1 sample (Figure 6B), the maximum grey values were set to 3360, 3380, 3400, 3410, 3430, 3440 (K-means clustering), 3450, 3470, 3490, and 3662 (Otsu method). The resulting segmentations included 3 phases: background (with grey values < 10), fracture (grey values between 10 and the maximum threshold), and matrix (grey values bigger than the maximum threshold).

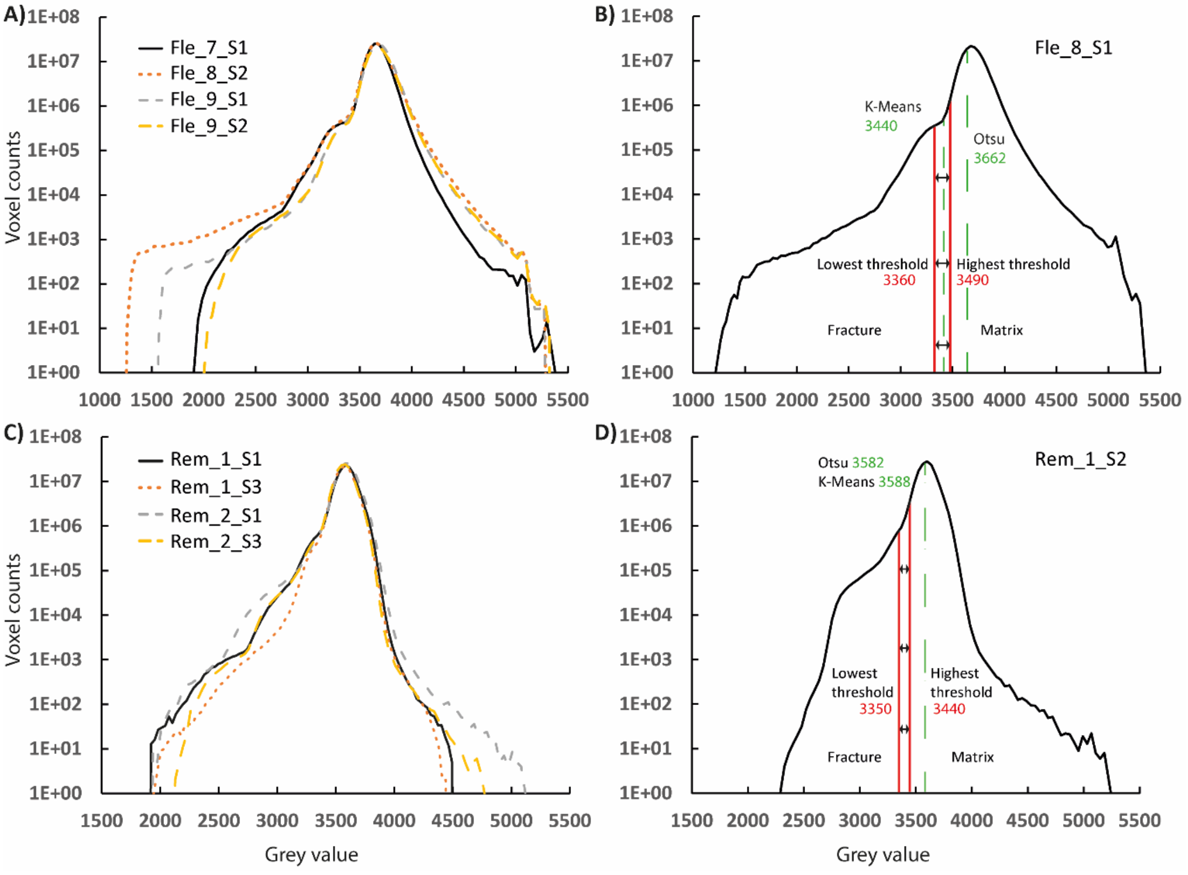


Figure 6. The grey value histograms of the Flechtinger and Remlinger cores
*Two representative samples (Fle_8_S1 **(B)** and Rem_1_S2 **(D)**) were chosen to quantify the quality of the applied segmentation method. The dotted green lines mark the thresholds obtained by the Otsu method and K-means clustering. The red lines mark the lowest and highest manual thresholds applied. Voxels with grey values between 10 and the threshold value were segmented as fracture (pore) voxels, while voxels with a greater grey value were assigned as impermeable matrix (solid) voxels. The grey value histograms of the other samples are presented in **(A)** and **(C)**. (B) and (D) are adapted after Reinhardt et al. (2022).*

Seeds for the watershed segmentation were placed in the grey value images by manually applying one threshold for each phase. Voxels within these ranges were selected as seeds for the corresponding phase. For the Fle_8_S1, thresholds were set to 0 – 1177 (for the background phase), 1177 – 3350.4 (for the fracture phase), and 3630 – 5344 (for the matrix phase). Seeds for the Rem_1_S2 were placed into voxels with grey values between 0 – 130 (background), 131 – 3310 (fracture) and 3800 – 5430 (matrix). Each threshold was selected to avoid falsely segmenting voxels near the boundary between fracture and matrix (and thus avoiding over-segmenting the fracture) while segmenting as many fracture voxels as possible. After placing the seeds, the region growing algorithm, implemented in Avizo 2019.3 (based on Beucher & Meyer 1993), was applied.

In order to remove matrix voxels that were falsely segmented as fracture voxels (islands), the "cleanse" tool (implemented in GeoDict 2020) was applied. This tool searches for fracture components connected by face, edge, or vertex and reassigns their phase to the matrix phase if their size is smaller than a given value. Additionally, the datasets were cropped to remove ring artefacts (see subsection 2.2.2) on the upper and lower end (in the Z-direction). The Fle_8_S1 was cropped by 50 slices on both ends, resulting in dimensions of 570 x 570 x 689 voxels. The Rem_1_S2 sample was cropped to dimensions of 570 x 570 x 679 voxels (50 slices at the upper and 60 slices at the lower end). Figure 7 illustrates the segmentation procedure.

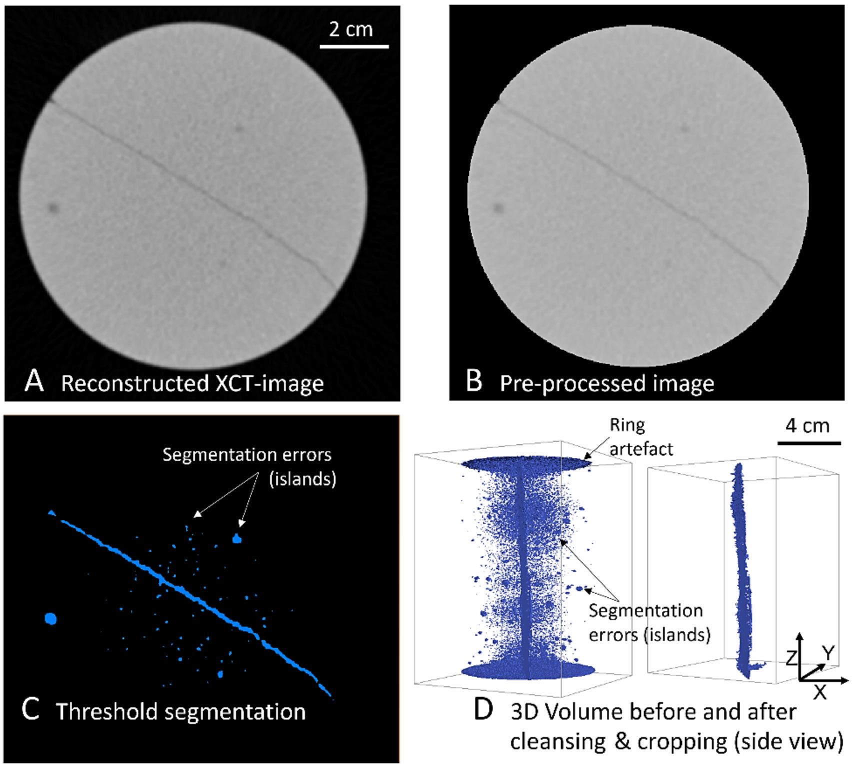

2 cm
A Reconstructed XCT-image
B Pre-processed image
Segmentation errors (islands)
C Threshold segmentation
Ring artefact
4 cm
Segmentation errors (islands)
Z
Y
X
D 3D Volume before and after cleansing & cropping (side view)

Figure 7. The segmentation procedure is illustrated on the Rem_1_S2

*(**A**) The reconstructed grey value image. **(B)** Masked pre-processed image, after resampling to a resolution of 200 µm, changing the image type to 16-bit unsigned, and removing negative grey values. **(C)** Segmentation using a manual global threshold of 3420. Fracture (pore) voxels are shown in blue, matrix, and background voxels in black. **(D)** Errors were removed by 1) Cropping top and bottom slices of the dataset (removes ring artefacts) 2) Applying the cleanse tool (removes falsely segmented voxels ("islands")). Adapted after Reinhardt et al. 2022.*

3.3.2 Convolutional neural network

This section describes the CNN used for segmenting both representative samples (Fle_8_S1 and Rem_1_S2). The CNN was adapted, optimized, trained, and applied by Asst. Prof. Dr. Saeid Sadeghnejad.

Both samples (Fle_8_S1 and Rem_1_S2) were segmented using a CNN based on a U-net-like architecture (Ronneberger et al. 2015). An encoder based on the form proposed by Simonyan and Zisserman (2015) was used. The architecture of the applied network is presented in Figure 8. The structure is divided into two parts: 1) the encoder modules extract image features by down-scaling the image resolution, and 2) the decoder modules turn the extracted features into predictions by up-scaling. Encoder layers (composed of several blocks) are linked with their equivalent decoder layer in the U-net structure by using symmetric skip connections. These connections allow using extracted shallow features in the decoding layer, which increases training accuracy (Wang et al. 2021b).

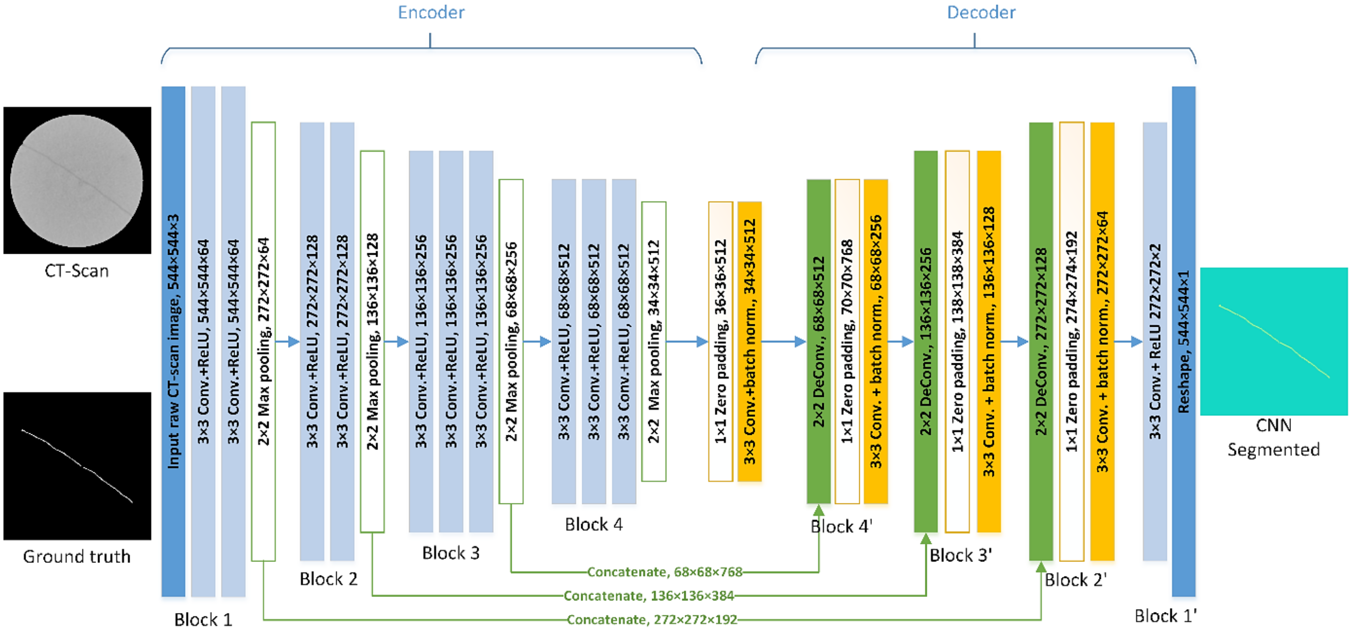

Encoder
Decoder
CT-Scan
Ground truth
Input raw CT-scan image, 544×544×3
3×3 Conv.+ReLU, 544×544×64
3×3 Conv.+ReLU, 544×544×64
2×2 Max pooling, 272×272×64
3×3 Conv.+ReLU, 272×272×128
3×3 Conv.+ReLU, 272×272×128
2×2 Max pooling, 136×136×128
3×3 Conv.+ReLU, 136×136×256
3×3 Conv.+ReLU, 136×136×256
3×3 Conv.+ReLU, 136×136×256
2×2 Max pooling, 68×68×256
3×3 Conv.+ReLU, 68×68×512
3×3 Conv.+ReLU, 68×68×512
3×3 Conv.+ReLU, 68×68×512
2×2 Max pooling, 34×34×512
1×1 Zero padding, 36×36×512
3×3 Conv.+batch norm., 34×34×512
2×2 DeConv., 68×68×512
1×1 Zero padding, 70×70×768
3×3 Conv. + batch norm., 68×68×256
2×2 DeConv., 136×136×256
1×1 Zero padding, 138×138×384
3×3 Conv. + batch norm., 136×136×128
2×2 DeConv., 272×272×128
1×1 Zero padding, 274×274×192
3×3 Conv. + batch norm., 272×272×64
3×3 Conv.+ ReLU 272×272×2
Reshape, 544×544×1
CNN Segmented
Block 1
Block 2
Block 3
Block 4
Block 4'
Block 3'
Block 2'
Block 1'
Concatenate, 68×68×768
Concatenate, 136×136×384
Concatenate, 272×272×192

Figure 8. Architecture of the implemented U-net CNN
Greyscale XCT-images are used to create segmented images with two phases (fracture and matrix/background). Reinhardt et al. 2022.

The encoder blocks consist of (i) convolutional layers with a kernel size of 3 x 3. These layers consist of a set of filters with different weights, which can be learned during training. Feature maps are created by applying convolution calculations (using filters) with the input images. The non-linear activation function (Rectifier linear unit (ReLU)) is applied to allow the CNN to represent non-linear features and to speed up the training procedure. The ReLU returns zero for negative inputs and the input value for positive inputs. (ii) The max-pooling layers have kernel sizes of 2 x 2 (with a stride of two). These layers reduce the size of the input images by a factor of two (by selecting the largest element in each pooling region), which reduces computational demands, avoids overfitting, and enhances the ability of the CNN to represent non-linearity (Hong and Liu 2020).

The decoding blocks consist of a deconvolutional layer (which doubles the size of the feature maps), a 1 x 1 zero padding layer (ensuring that the centre of each filter corresponds to the first pixel of the input images), a convolution layer with batch normalization (stabilizing the network and speeding up the training process by standardizing the inputs, (Ioffe and Szegedy 2015)). The last layer is a convolution layer with two filters to discriminate between fracture and matrix/background phase.

The segmentation output is assessed by applying a cost function. A typical example is the binary cross entropy (BCE, Shelhamer et al. 2017; Siam et al. 2018), which is a measure of the similarity between the ground truth (g_t) and the predicted segmentation (s_p). The BCE is calculated with equation 17:

$$BCE = -g_t \log(s_p) - (1 - g_t)\log(1 - s_p) \tag{17}$$

However, the BCE can be an unsuited measure for highly unbalanced images. For instance, in a binary image with a low proportion of one phase (e.g., fracture pixels) to the other one (e.g., background pixels), the BCE could favour a predicted segmentation that simply labels the whole image as background. Common metrics that allow dealing with unbalanced class distributions in image segmentation are the Jaccard index and Jaccard distance (e.g., see Yuan et al. 2017; Siam et al. 2018; Tang et al. 2019; Van Beers et al. 2019). The Jaccard distance loss (J_l) is a measure for the dissimilarity between two images (of s_p and g_t) and is expressed in equation 18:

$$J_l = 1 - IoU = 1 - \frac{|s_p \cap g_t|}{|s_p| + |g_t| - |s_p \cap g_t|} \tag{18}$$

The intersection over union (IoU, also called the Jaccard index) is calculated by dividing the overlapping area of two images ($|s_p \cap g_t|$) by their union area ($|s_p \cup g_t| = |s_p| + |g_t| - |s_p \cap g_t|$). The IoU index ranges from one (both images are identical) to zero (the images do not overlap, i.e., they are completely dissimilar).

Since the fracture pixels occupy only a small fraction of each 2-D image (~ 1 % fracture vs ~ 99 % matrix/background pixels), a weighted loss function (L), combining BCE and J_l, is applied to calculate the similarity between the predicted segmentation and the ground truth.

$$L = wJ_l + (1 - w)BCE \tag{19}$$

with a weight (w) of 0.8.

The ground truth is crucial for training a CNN. A common practice is the preparation of a manually labelled ground truth, which is time-consuming and prone to errors from user input. Therefore, the ground truth for the employed CNN was based on the segmentation output of the RF classifier. Beforehand, the binary images from the RF classifier were corrected by reassigning matrix voxels that were falsely segmented as fracture voxels (with the cleanse tool implemented in GeoDict 2020). The image stacks were composed of 679 (Rem_1_S2) and 689 (Fle_8_S1) 2-D images with a size of 544×544 pixels. In order to increase the accuracy of the prediction, the number of training data was increased by applying geometrical modifications to the original data. These data augmentations were applied by using the *imgaug* library (https://github.com/aleju/imgaug) and include rotating by -45 to +45°, scaling to 80 to 120 %, mirroring, cropping from 10 to 30 %, shearing by -15 to +15°, and translating by -20 to +20 %. Training of the network was performed on an Nvidia GeForce GTX 1080 Ti GPU with 11 GB RAM (random-access memory) and realized with an RMSProp optimizer, a gradient-based optimization method (Hinton et al. 2012). An initial learning rate of 1e-5 was used.

To show the generality of this method, training of a network and applying the network (to obtain a segmented dataset) were not conducted on the same sample. The CNN was trained on the augmented images of one sample (e.g., Fle_8_S1). After the training phase was completed, the CNN was tested on the 16-bit greyscale XCT-images of the other sample (e.g., Rem_1_S2). Afterwards, the sample input was switched (i.e., the network was trained on the Rem_1_S2 and applied on the Fle_8_S1). 80 % of the augmented images were used for the training and 20 % for the subsequent evaluation.

3.3.3 Random forest classifier

Due to its robustness and superior generalization performance, the RF classifier was selected as the second machine learning segmentation method (Berg et al. 2019). In order to conduct the segmentation, the RF implementation of the Ilastik software (version 1.3.3) was used since it offers a predefined, customizable workflow that was successfully applied in previous studies (e.g., Eyolfson et al. 2020; Brady et al. 2020; Babel et al. 2020). A set of filters (Gaussian Gradient Magnitude, Laplacian of Gaussian, Gaussian Smoothing, Difference of Gaussians, Structure Tensor Eigenvalues, and Hessian of Gaussian Eigenvalues) with varying parameters is chosen to create the features used by the RF classifier to segment the input data. Afterwards, training data is provided by manually marking voxels of each phase in one slice at a time. This approach lowers the computational demands and allows working on datasets that are significantly larger than the RAM (Berg et al. 2019). Conducting the training on 3-D images is possible but has a higher computational demand. During the training phase, preliminary results and uncertainties of the segmentation can be used to further improve to provided training data. The software can handle input data in up to 5-D (3D, channels, and time).

3.4 PVE Correction

The common methods applied for PVE correction in fractures and a discussion of their disadvantages can be found in section 2.3. The calibration-free missing attenuation (CFMA) method (Huo et al. 2016) was applied in this study since calibration data (which is needed for other approaches) was not available, and the CFMA method was shown to be suitable for determining small apertures, even below the image resolution (Ketcham et al. 2010; Huo et al. 2016). Although the approach of Cappuccio et al. (2020) also seems promising for conducting a fracture reconstruction, they used better resolved fractures for their calibration and their study, compared to the samples in this analysis. Due to the small fracture apertures and the low image resolution, a large influence of the PVE on the samples used in this study was expected. Therefore, this thesis evaluated if the CFMA method is suitable for conducting a proper fracture reconstruction with enhancing the spatial resolution. Similar fracture reconstructions were done before (e.g., Ramandi et al. 2017); however, these approaches require additional information from calibration data (e.g., from SEM images). In contrast to the fracture reconstruction conducted by Ramandi et al. (2017), medical XCT images are used, i.e., the analysed samples are bigger and, thus, the scanned resolution is distinctively different from the µXCT images (around 200 µm vs approximately 16.5 µm). Furthermore, the reconstructed fractures were validated with various parameters instead of only using the permeability (as in the study by Ramandi et al. 2017).

After evaluating the best segmentation method for the measured XCT-images, the resulting binary images can be used to correct the images for the PVE. Figure 11 presents the developed workflow to reconstruct a single fracture at a higher resolution, using the grey value XCT-images (Figure 11A) and binary images obtained by the segmentation. At present, the CFMA method (after Huo et al. 2016) was not applied for reconstructing fractures in higher resolutions. Applications of the CFMA include the calculation of fracture volumes and the reconstruction of fractures based on additional calibration data and without enhancing the spatial resolution (e.g., Kling et al. 2016; Shen et al. 2021). The workflow presents a novel approach by applying the CFMA method for reconstructing single fractures at a higher resolution without requiring any calibration data. The results from the presented analysis are a first step in evaluating the applicability of the CFMA method for correcting the PVE to obtain a more realistic DT. This section also discusses the limitation of this study. Potential approaches for future studies are given in subsection 4.2.4. In order to allow for further evaluation and improvement of the presented workflow, the code is open source and can be found in the appendix (Figure 43 - 46).

The principle of the CFMA method is based on the comparison between measured CT values of voxels from the rock matrix, fracture, and background. Voxels from the background and the matrix have the minimum and maximum CT values since the former voxels are not affected by the sample, and the latter voxels are fully affected by the presence of the rock. Depending on the size of present fractures and the resolution of CT measurements, voxels belonging to fractures are usually a mixture of rock and void greyscale values (in varying portions) and have then lower CT values (compared to voxels of only rock matrix). This difference is called the missing attenuation. In the CFMA method, profiles are locally taken across the fractures (ideally perpendicular to the fracture orientation if the goal is the calculation of true local apertures). CT values of voxels in these profiles are compared to CT values of the rock matrix. The difference between these values can be used to calculate the local (apparent or true) aperture of fractures. The CFMA method works well for fractures with a small aperture compared to the spatial resolution. In this case, the fracture volume is highly smeared, and information about the missing attenuation is distributed over more voxels. However, the profiles are chosen to locally include each voxel that is affected by the presence of fractures. The information of each voxel in a profile is compared to the CT values of voxels from the rock matrix and summed up (Ketcham et al. 2010; Huo et al. 2016).

Figure 9 illustrates the principle of the CFMA method and the basic idea of the PVE correction workflow. In this simple case, only one pixel (with a spatial resolution of 200 µm) is considered. The grey value of the pixel is a mixture between the grey value of the matrix and the fracture (Figure 9 A). The aperture of the fracture is below the spatial resolution and cannot be accurately represented in this resolution. The CFMA method is applied to this pixel to quantify the influence of the fracture on the pixel's grey value. In this way, the true fracture aperture is calculated as 100 µm (the CT values of the voxel, air, and matrix are used in this calculation). In the next step, the original pixel is resampled to a spatial resolution of 50 µm (in a grid with 16 smaller pixels), which is sufficient to resolve this fracture aperture. Then, the fracture skeleton (here, the medial axis of the resampled pixel) is calculated and represented in the new grid (Figure 9 B). This skeleton is grown in the final step. The information about the true local aperture (100 µm, obtained in the first step) is used to determine the extent of the fracture growth. In this example, the true aperture can be represented with two pixels of 50 µm size. Hence, the skeleton is grown by one pixel in each column of the grid (Figure 9 C). The final result is an image in a higher spatial resolution, where the fracture is accurately reconstructed. In the original image, the fracture is a sub-resolution feature and could, therefore, not be extracted by a segmentation procedure.

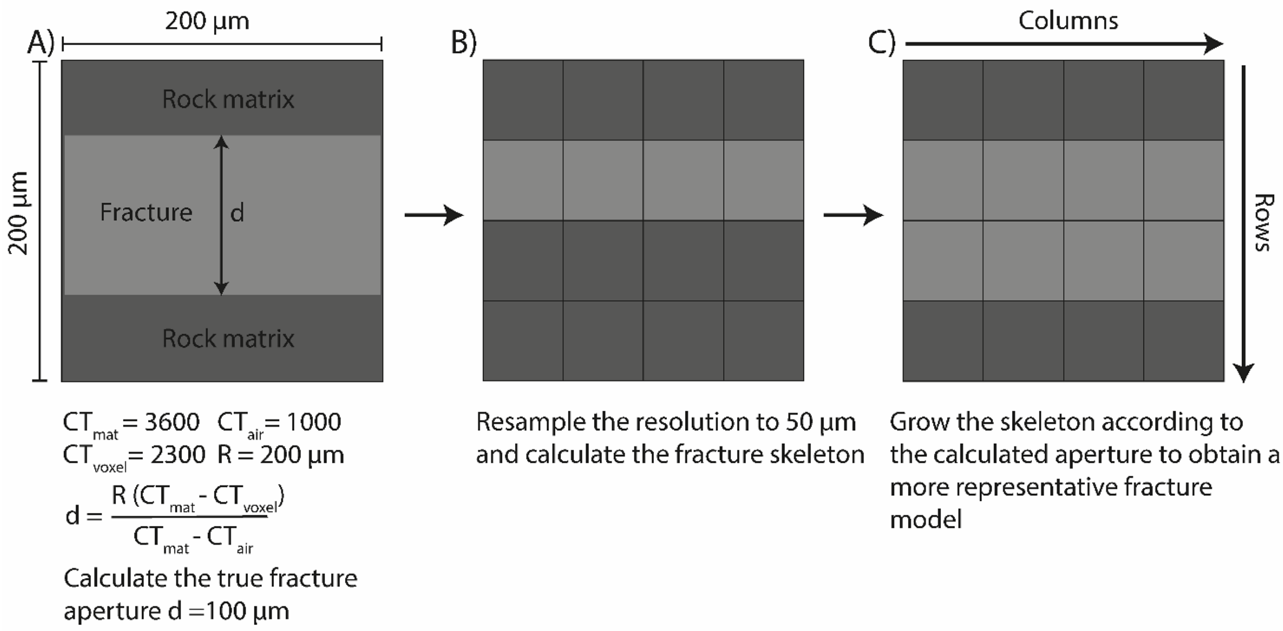

A)
200 µm
200 µm
Rock matrix
Fracture
d
Rock matrix
$CT_{mat} = 3600$ $CT_{air} = 1000$
$CT_{voxel} = 2300$ $R = 200$ µm
$d = \frac{R\,(CT_{mat} - CT_{voxel})}{CT_{mat} - CT_{air}}$
Calculate the true fracture aperture d =100 µm
B)
Resample the resolution to 50 µm and calculate the fracture skeleton
C)
Columns
Rows
Grow the skeleton according to the calculated aperture to obtain a more representative fracture model

Figure 9. Illustration of the principles for the applied fracture reconstruction
*(**A**) The illustrated pixel (200 µm x 200 µm) contains two phases: matrix and fracture; thus, the resulting grey value of the pixel is a mixture of the grey values of both phases. The CFMA method is applied to calculate the true fracture aperture, which is below the resolution of 200 µm. **(B)** The pixel depicted in (A) is resampled by a factor of 4, and the fracture skeleton is calculated. **(C)** The skeleton is extended by considering the true fracture aperture, calculated in the first step. In this case, the fracture skeleton is extended by 1 pixel in each column to match the aperture of the reconstructed fracture with the real fracture aperture of 100 µm.*

In the first step of the applied PVE correction workflow, the CT values for air (CT_{air}) and the global rock matrix (CT_{mat}) are determined with the pre-processed greyscale XCT-images (the grey value images of the analysed samples were processed by: converting the image type to 16-bit unsigned, resampling to a resolution of 200 µm and shifting the grey values by 2000) using Avizo 7.0.1 (VSG, France). In order to exclude the influence of irrelevant pixels (only background voxels are considered for the CT_{air} value, and only pixels from the rock matrix are considered for the global CT_{mat} value), the images are masked (i.e., setting the grey value of each irrelevant pixel to zero) prior to the calculation of the mean CT values. The mean CT_{air} value is obtained by taking the arithmetic mean of the grey value of all background pixels, and the global CT_{mat} values are calculated by using the arithmetic mean of the grey values of all matrix pixels (excluding pixels in the background, at the boundary between background and matrix, and in the vicinity of the fracture). Note, since the grey values in the XCT-images were shifted by 2000, the calculated CT_{air} and CT_{mat} values must be subtracted by 2000 to allow a comparison with CT values reported in the literature.

In the second step, the segmented fracture (obtained with the RF classifier) is processed using the 3-D visualization software Avizo 7.0.1: (i) matrix pixels that are falsely classified as fracture pixels are removed by applying the “islands remove” algorithm in 3D. This algorithm searches for connected pixels of a specific phase and reassigns their phase if the size of the connected area is smaller than a given threshold. (ii) The segmentation is extended by 4 pixels in each direction by using the “grow” algorithm four times. This step depends on the segmentation, samples, and spatial resolution. Hence, it is not necessary to always grow the segmentation by 4 pixels; in some cases, more or fewer pixels have to be added to include the whole fracture area. (iii) Since the segmented fracture area is extended in all directions in the previous step, the segmentation at the edges of the core overlaps with the background (i.e., surpasses the core edge). In order to remove these errors, the mask applied on the pre-processed images (see section 3.2) is applied for the second time. Thus, the result is the segmented fracture area without segmentation errors in the matrix and extended by 4 pixels. Since the PVE could lead to a significantly blurred fracture in the grey value images, these steps ensure that the whole area, which is influenced by the presence of the fracture, is segmented. This includes the boundary between fracture and matrix, where the RF segmentation shows the highest uncertainties (Figure 18C). Thus, uncertainties from the segmentation are further minimized. The resulting binary dataset is multiplied with the pre-processed XCT-images to produce grey value images (Figure 11C) of the isolated fracture area (i.e., the matrix and background are set to a grey value of zero). These images are exported as Tif-files for the subsequent analysis. Figure 10 illustrates the applied workflow to reconstruct the fractures at a higher resolution.

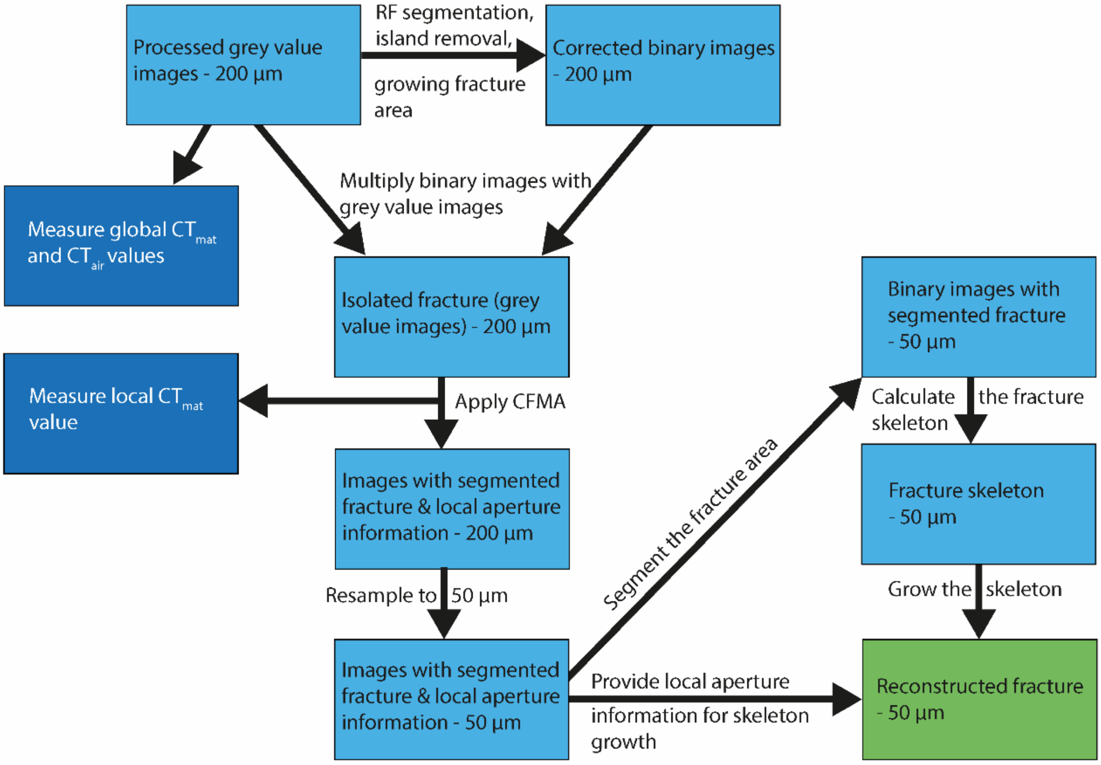

Processed grey value images - 200 µm
RF segmentation, island removal, growing fracture area
Corrected binary images - 200 µm
Measure global CT$_{mat}$ and CT$_{air}$ values
Multiply binary images with grey value images
Isolated fracture (grey value images) - 200 µm
Measure local CT$_{mat}$ value
Apply CFMA
Images with segmented fracture & local aperture information - 200 µm
Resample to 50 µm
Images with segmented fracture & local aperture information - 50 µm
Segment the fracture area
Binary images with segmented fracture - 50 µm
Calculate the fracture skeleton
Fracture skeleton - 50 µm
Grow the skeleton
Provide local aperture information for skeleton growth
Reconstructed fracture - 50 µm

Figure 10. Flow chart of the applied reconstruction workflow

In the next step, the local apparent (vertical) apertures and local CT$_{mat}$ values are measured by applying a self-developed python (version 3.8.8) code. The images are imported with the Tif-file python library (https://pypi.org/project/tifffile/) and treated as matrixes containing grey values of the fractured area. The algorithm iterates over each column in each image matrix. In a column, the algorithm searches for non-zero values in each row. Since only single fractured rocks are considered, and the fractures are isolated by applying a mask in the previous step (i.e., the grey values of matrix and background pixels were set to zero), the fracture is identified in each column of an image matrix by finding the non-zero pixels. These pixels are used to calculate the local apparent (vertical) aperture in each column. In this way, profiles (parallel to the Y-axis) are created across the fracture (an exemplary profile is shown in red in Figure 11D). The following procedure is executed for the first profile and then subsequently applied to the following profiles until the last column of the current image matrix is reached. Then the procedure is applied to the next image until the image stack is fully analysed.

A profile spans from the first non-zero pixel to the last non-zero pixel in the currently analysed column (i.e., only considering pixels of the isolated fracture area; Figure 11D). Firstly, the local CT$_{mat}$ value in the column is calculated by taking the arithmetic mean of the grey values of ten pixels, which are assumed to belong to the matrix: the first and last pixel in the profile and the four pixels above, and the four pixels below the profile. Since these pixels are located outside the segmented fracture and therefore contain a grey value of zero, the corresponding grey values are retrieved from the pre-processed grey value XCT-images (Figure 11A). The apparent local aperture is calculated according to equation 20 by considering the second (see the red profile in Figure 11D, i = 1) to the next to last pixels (i = N_{vox}) in the currently analysed profile:

$$d = \frac{R\sum_{i=1}^{Nvox}(CT_{mat} - CT_i)}{CT_{mat} - CT_{air}} \qquad (20)$$

with the fracture aperture (d), the resolution (R), the CT values of air (CT_{air}), the matrix (CT_{mat}, which can be determined globally or locally), and the voxel influenced by the presence of the fracture (CT_i).

A pixel is only considered in the calculation if the grey value is lower than the calculated CT_{mat} value (i.e., that the grey value of the pixel is influenced by the presence of the fracture). The positions of the first and last pixel to be considered in the calculation are used to determine the centre pixel of the fracture in the current profile. The calculated local apparent aperture is stored in a new image stack (having the same dimensions as the input images) at the position of this centre pixel. Since the centre pixels of each column are not necessarily connected to each other, fracture pixels are added above and below the centre pixels to ensure that a connected fracture area, which is required for the next steps, is created. The fracture centres are grown according to the apparent aperture value by storing the apparent aperture value alternately in the pixels above and below the centre pixel. For example, apparent aperture values below 75 µm do not result in growth, up to 125 µm result in also storing the apparent aperture value in the pixel below the centre pixel, up to 175 µm result in storing the value in the pixel below and above the centre pixel, up to 225 µm result in storing the value in the next two pixels below and one pixel above the centre pixel, up to 275 µm result in storing the value in the next two pixels below and the next two pixels above the centre pixel, etc. Apparent aperture values above 375 µm are stored in seven pixels around the centre. The growing steps of 50 µm are selected by considering the aperture distribution of the fractures (also see section 4.2, most measured apertures are smaller than 400 µm) and due to the similarity to the subsequent growing procedure of the resampled fractures (with a spatial resolution of 50 µm). In that way, a thicker skeleton (containing the measured aperture values) of the segmentation is created, ensuring that the fracture connectivity and shape (i.e., small fracture sections remain small and large fracture sections remain larger in the grown skeleton) are preserved during the subsequent steps (i.e., resampling and skeletonizing of the resampled fracture).

Figure 11E presents a close-up after finishing the analysis for each profile in each column and image. Afterwards, the image stack is resampled to a resolution of 50 µm using the box filter implemented in Avizo 7.0.1. The images are segmented (Figure 11F) by using a manual threshold with a range of 1 to the maximum apparent aperture measured. The resulting binary images contain two phases: fracture and matrix/background. These images are used to calculate the skeleton of the fracture by applying the bwscel function, implemented in Matlab R2021a (https://de.mathworks.com/help/images/ref/bwskel.html). The bwscel function is applied in 2-D to reduce the fracture in each image to a single line (Figure 11G). The minimum branch length is set to 25 voxels to avoid unwanted branches. This skeleton is grown in the last step, similar to the procedure described above. The skeleton is grown in the vertical direction (parallel to the Y-axis) according to the apparent aperture stored in the image stack created during the apparent aperture calculation step (Figure 11E). For example, with a growing limit of 75 µm, the skeleton does not grow at locations with apparent apertures smaller than 75 µm; one pixel is added to the skeleton at locations with apparent apertures of 75 – 125 µm, etc. The maximum growth for the presented samples is 41 pixels, corresponding to the highest local apparent aperture (2035 µm) calculated for the fractures. In that way, the apparent local aperture of the reconstructed fracture is adjusted to match the apparent local aperture calculated from the grey value XCT-images (Figure 11H).

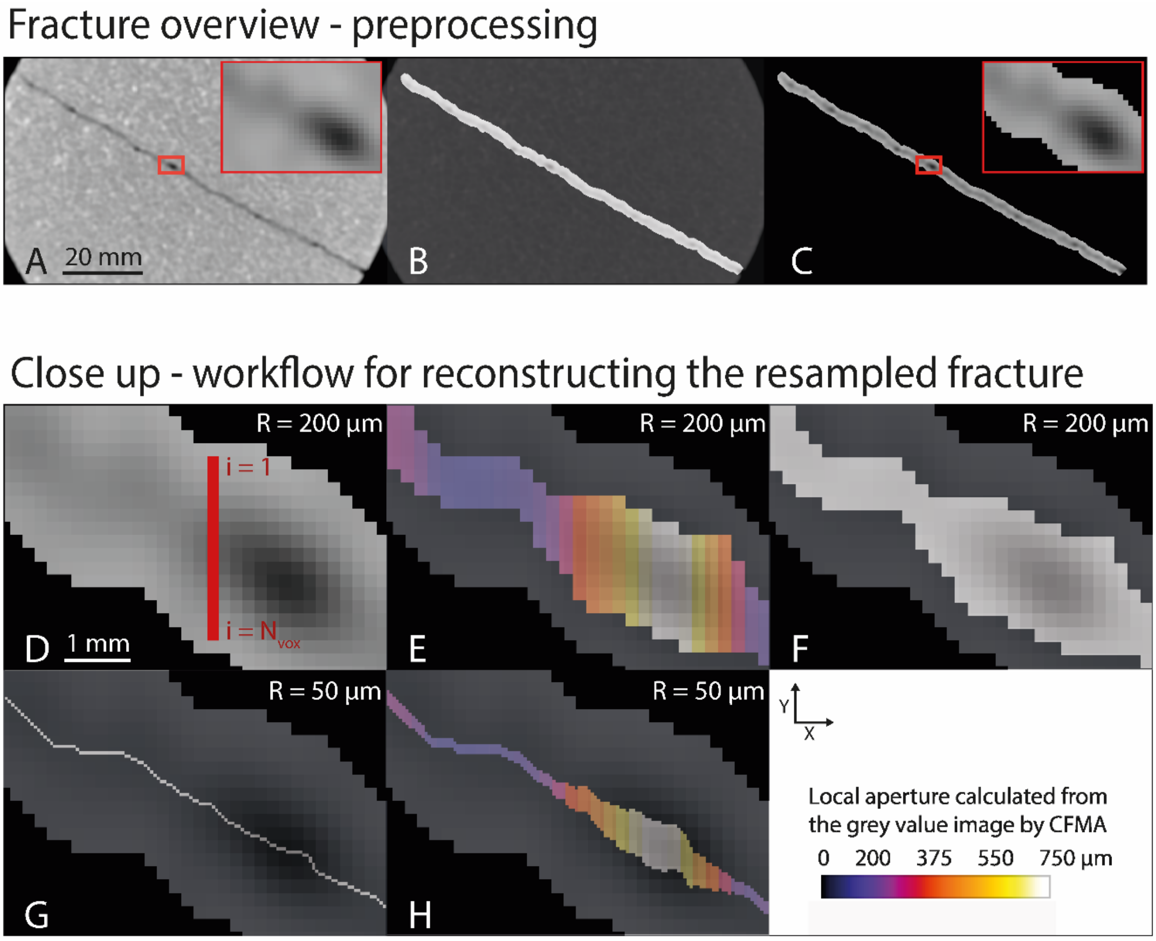

Fracture overview - preprocessing
A
20 mm
B
C
Close up - workflow for reconstructing the resampled fracture
R = 200 µm
i = 1
i = N_vox
D
1 mm
R = 200 µm
E
R = 200 µm
F
R = 50 µm
G
R = 50 µm
H
Y
X
Local aperture calculated from the grey value image by CFMA
0 200 375 550 750 µm

Figure 11. Workflow to reconstruct the fracture from the XCT images
***(A)** Pre-processed grey value XCT images **(B)** Segmented fracture, grown by four voxels in X and Y direction to ensure that each voxel whose grey value is influenced by the fracture is included. The grey value image (A) is shown in the background. **(C)** Result of multiplying the grey value images (A) with the segmentation (A). Pixels in the vicinity of the fracture contain the measured grey values; the other pixels are assigned a value of zero. The close-up (framed in red) is shown in the subsequent panels (D) – (H) to illustrate the further workflow. **(D)** In the close-up shown in (C), the red line shows an exemplary profile that is used to measure the apparent aperture in the next step. In each column, a profile along the non-zero pixels was created (parallel to the Y-axis). The first and last pixel of the profile and eight pixels (four prior and four after the profile) outside the segmented area were used to locally calculate the* CT_{mat} *value. Along the profile (red line), grey values greater than the locally calculated* CT_{mat} *value were used (*$i = 1$ *to* $i = N_{vox}$*) to calculate the apparent local aperture. **(E)** Result after applying the apparent aperture measurement and storing the apparent aperture values in a new image stack. **(F)** The image stack containing the apparent aperture values (E) is resampled to an image resolution of 50 µm and segmented. These images are then used to calculate the skeleton **(G)** of the structure. **(H)** The skeleton is grown in Y-direction according to the measured apparent aperture stored in (E). The result is a reconstruction of the fracture in 50 µm. Note: the aperture values in (E) and (H) are the calculated local, true fracture apertures, not the apertures of the segmentation (E) or the reconstructed fracture (H). The calculation of the mechanical apertures of the reconstructions is discussed in subsection 3.5.3.*

The result is a reconstructed fracture with 50 µm resolution, based on the grey value information of the original XCT-images (with a resolution of 200 µm). The workflow can be adapted (e.g., by including other methods like a 3-D skeletonization/medial axis computation or different resampling methods) to optimize the analysis for other studies. The error sources for the reconstructions presented in this study include: (i) the local apparent aperture calculation with the CFMA method is known to have uncertainties. For example, Huo et al. (2016) and Wenning et al. (2019) reported deviations of about 20 – 60 µm for their fractures. Furthermore, Kling et al. (2016) showed that minor deviations in the calculated CT value of the matrix can have a significant influence on the reconstructed structure (ii) the measured apparent aperture could only be applied in 50 µm steps (limited by the computational power which allowed a maximum resampled resolution of 50 µm). Since the resolution was still in the range of the real fracture apertures, it was not possible to identically represent the real aperture heterogeneity and small-scale roughness. In order to account for these errors, various reconstructions were created by varying the following two parameters:

(i) Instead of local CT_{mat} values, a global CT_{mat} value was used.

(ii) The thresholds for growing the skeleton in the last step were adjusted. Various cases were considered where apparent aperture values < 35 - 120 µm did not result in the local growth of the skeleton. The subsequent limits followed in 50 µm steps. That means the skeleton was grown by adding a pixel for each 50 µm increase in apparent local aperture (starting from the minimum value (referred to as the growing limit in the subsequent chapters) of e.g., 75 µm up to 2125 µm). The maximum local growth applied in this study was 41 pixels, which corresponds to the highest apparent aperture calculated for the presented fractures.

These parameters were adjusted to achieve the best possible fit between the experimental data with the simulated breakthrough curves using the reconstructed structures. The resulting fracture models were therefore calibrated with the most sensitive data available and can be considered to reliably represent the real fractures.

The algorithm presented in this section calculates the vertical local apparent apertures of the fractures. The vertical apertures are calculated because the orientation of the here analysed fractures is horizontally (parallel to the X-axis) with inclinations (depending on the sample) between about 5 - 40°. Alternatively, the horizontal local apparent apertures could be calculated for fractures that are orientated vertically.

3.5 Validation parameters

Validation poses a crucial step in the DRP workflow to be able to conclude from simulation results. It must be ensured that the processed structures represent reality. Usually, more than one parameter is required to conduct a proper validation (Leu et al. 2014; Berg et al. 2018). The following subsection presents the parameters selected to validate the DT models with the real fractures.

3.5.1 Water injection and tracer experiments

This subsection describes the experimental procedure for determining the hydraulic aperture, estimated fracture volumes, permeability, and breakthrough curves for the fractured samples. These laboratory experiments were conducted by the ReSalt project partners at the RUB. Further details are described in Frank et al. (2020b, a).

In order to prevent displacement of the core halves during the experiments, the spitted cores were encased in a thin-walled, stable shrink tubing with overlapping ends at the lower and upper ends of the cores. Air inclusions in the cores were removed using a desiccator connected to a vacuum pump. Afterwards, the cores were saturated with water. The saturated cores were mounted in a self-constructed clamping device. The inflow and outflow were symmetrically designed, each consisting of a Teflon attachment with a hose connection. A sealing ring between the Teflon attachment and the core ensured that the system was sealed. The sealing ring also prevented bypassing fluid flow since it was pressed against the overlapping ends of the shrinking tube. This design ensured that fluid flow only occurred through the fractures (Figure 15).

A constant hydraulic gradient was applied to enable water flow from the bottom of the core to the top. The flow rate was measured at the outflow, and water samples were taken. In order to calculate the hydraulic parameters, Darcy's law must be applicable, i.e., it must be ensured that laminar flow conditions prevail. A standard measure to determine the flow regime is the Reynolds number, which is the ratio of inertial to viscous forces. For small Reynolds numbers, the flow can be considered laminar. For the flow in fractures, a critical Reynolds number of about 2400 was suggested for the transition from laminar to turbulent flow (Witherspoon et al. 1980; Dippenaar and Van Rooy 2016). The Reynolds number was calculated after equation 7 (see section 2.5) for each experiment and was found to be smaller than 1 (see Table 7 in subsection 4.2.2), which indicates laminar flow regimes. Furthermore, for these Reynolds numbers, the simplification of the Navier-Stokes equation to the Stokes equation to simulate the flow fields is justified (Schwarz and Enzmann 2013). The measured flow rate was used to determine the hydraulic aperture a_h of the fractures by applying a modified version of the cubic law (Tsang 1992; Becker 2008):

$$a_h = \sqrt[3]{\frac{12 \cdot \mu \cdot Q}{\rho \cdot g \cdot \frac{dh}{L_L} \cdot L_W}} \tag{21}$$

with the measured flow rate (Q, for the Fle 8 S1 = 1.14 mL/s; Rem 1 S2 = 0.3162 mL/s, see Table 1), the density of water (ρ) of 998.234 kg/m³ (at 20°C, after Kestin et al. 1978), the dynamic viscosity of water (μ) equal to 0.001 kg/m*s (at 20°C, after Kestin et al. 1978) and the estimated fracture width (L_w) of 0.099 m. A constant hydraulic gradient (i) of 4.7 [-] was applied by employing a hydraulic pressure gradient (dh) of 0.705 m via a water column over the core length (L_L) of 0.15 m. Estimated fracture volumes were calculated by multiplying the estimated dimensions of the fractures (estimated by using a width of 0.099 m and a length of 0.15 m, corresponding to the diameter and length of the cores, respectively) with the hydraulic aperture. Due to the estimation of the fracture dimensions and since the hydraulic aperture is usually smaller than the mechanical aperture (e.g., Xiong et al. 2011), the calculated fracture volumes are expected to be smaller than the real volumes. Equation 22 (de Marsily 1986) relates the hydraulic aperture with the intrinsic permeability (K) of the fracture:

$$K = \frac{a_{eff}^2}{12} \tag{22}$$

The transport behaviour of the fractures was analysed by conducting tracer experiments with the same flow settings as described above. 1 mL of a 2 molar NaCl tracer solution was injected as a pulse, lasting one second. A WTW Multi 3510 conductivity logger and a Tetracon 925c conductivity sensor were used to measure the electrical conductivity (EC) at the outflow. A measuring interval of one second was applied. For each sample, multiple tracer experiments were conducted to determine the reproducibility of the obtained breakthrough curves. Frank et al. (2020a) concluded that the deviations in the measured breakthrough curves can be attributed to the (aperture dependent) influence of surface roughness. More details of the procedure can be found in Frank et al. (2020b, a). The measured flow rates for each sample are summarized in Table 1.

Table 1. Measured flow rate for each flow-through experiment
Calculated permeabilities, Reynolds numbers and hydraulic apertures can be found in Figure 23 and Table 7.

Sample		Flow rate Q [mL/s]
Flechtinger	7_S1	1.06
	8_S1	1.13
	8_S2	0.70
	9_S1	1.04
	9_S2	0.73
Remlinger	1_S1	0.41
	1_S2	0.32
	1_S3	0.40
	2_S1	0.22
	2_S3	0.15

The measured EC values were converted into a fractional particle count to allow for a direct comparison with the simulated breakthrough curves. A linear relationship between EC and the number of particles in the water volume was assumed for the measured EC range (as reported by e.g., Kovačević et al. 2017; Quan et al. 2017; Martínez 2018; Frank et al. 2020a; Veerman 2020). Background EC values (measured by injecting water without adding the NaCl solution) were subtracted from the measured EC values. The corrected EC values from each timestep were divided by the sum of the corrected EC values of all timesteps. This conversion gave for each timestep the fraction of particles that were detected in the corresponding timestep, related to the sum of all particles detected at the outflow during the experiment.

3.5.2 Barite particle migration experiments

Over the last decades, the morphology of precipitated barium sulphate (barite) particles and their dependence on environmental conditions have been extensively studied. The variety of observed morphologies includes rounded, rhombic, plate, dendritic, hexagonal, and irregular shapes and depends, together with the size, on factors such as the concentration and ratio of ions (calcium, magnesium, barium, sulphate, pH-value), presence and type of additives (e.g., organic, inorganic, molecular structure, functional groups), formation processes (i.e., biotic or abiotic) or the temperature (Bertram and Cowen 1997; Qi et al. 2000; Hennessy and Graham 2002; Yu et al. 2005; Ruiz-Agudo et al. 2015b, a; Widanagamage et al. 2018). Due to the complexity of the barite formation, one main goal of the ReSalt project partner at the RUB was the in-depth analysis of the particle morphology in varying conditions. Barite particles were prepared and provided for further particle migration experiments in the HPTC.

In order to experimentally investigate barite migration through fractured rocks, the project partners at the RUB developed an HPTC, which is capable of simulating the whole geothermal production cycle (Frank et al. 2021). This includes in situ pressure and temperature conditions and the cooling of the circulating fluid before reinjection into the fractured core. The experiments were conducted on three samples (Fle_9_S1, Fle_9_S2, Rem_1_S2) by applying a temperature of 57 – 59°C in the core and a flow velocity of about 2 mL/s (Fle_9_S1) and 1.17 mL/s (Rem_1_S2). Since the pump malfunctioned during the experiment with the Fle_9_S2, the flow fluctuated between 0 and about 1 mL/s, which led to highly variable conditions during the experiment. Barite particles were introduced into the fractures by directly injecting a prepared barite particle mix in the upstream of the core (Fle_9_S1 and Rem_1_S2).

To equilibrate the setup, water circulated through the cores for one week before particles were injected. After this equilibration stage was completed, the barite particles were injected directly in front of the core and circulated for about one week. After a total runtime of 2 – 3 weeks, the HPTC was shut down, the cores dismounted, and one half of each fracture was imaged by a SEM machine. The variety of the deposited barite particles is presented in Figure 13. Shapes range from rounded (e.g., see the particles deposited in the Rem_1_S2) to flat, star-shaped particles. Details regarding the sample preparation, the HPTC and the experimental procedure on the Fle_9_S1 can be found in Frank et al. 2021. Additionally, the project partners at the TUD analysed the size distribution of the deposited barite particles (in the Fle_9_S2 fracture) by SEM (Figure 12). This allowed for an estimation of particle sizes used in the particle migration simulations.

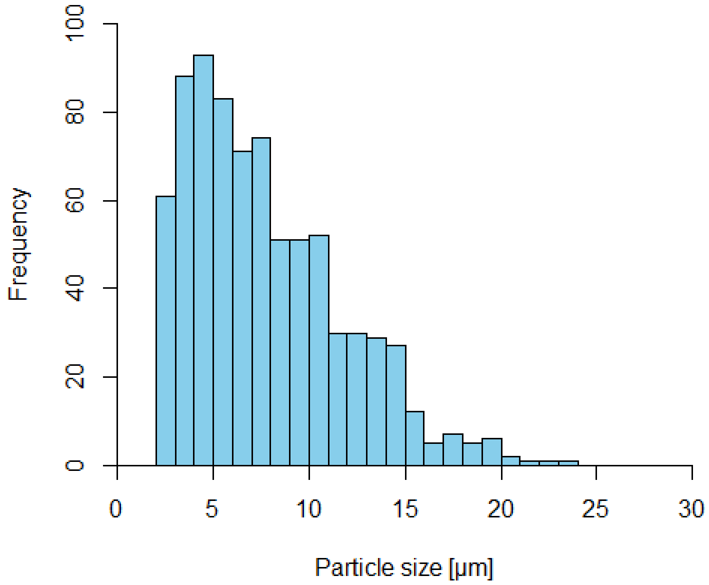


Figure 12. Size distribution of the barite particles deposited in the Fle_9_S2 fracture
780 particles were analysed. Prof. Dr. Martin Ebert from the TUD conducted the SEM analysis and provided the size distribution.

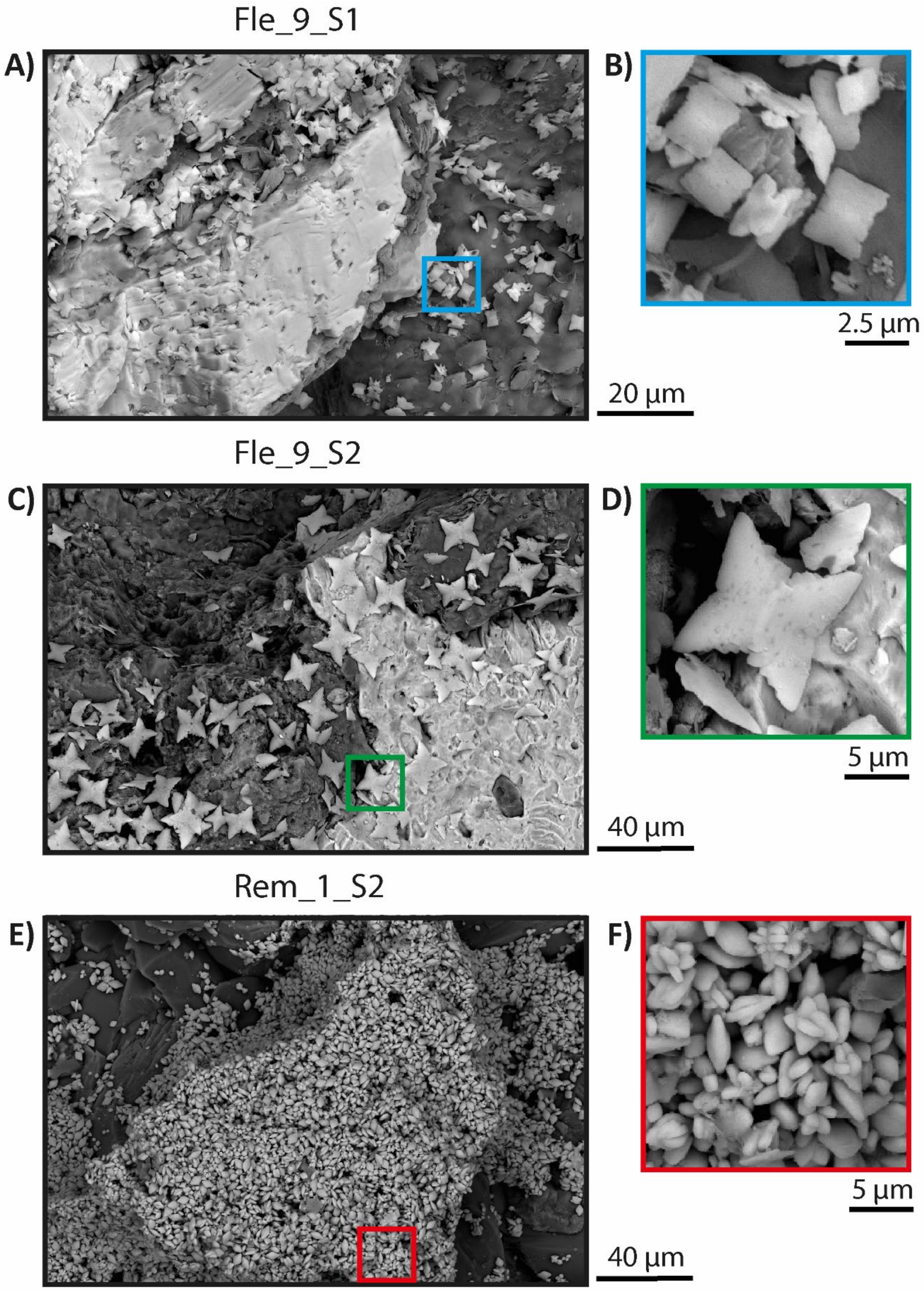


Figure 13. SEM snapshots of the deposited barite particles
In the experiments, particles were injected right in front of the fractured cores: Fle_9_S1 ***(A)****, Fle_9_S2* ***(C)*** *and Rem_1_S2* ***(E)****. Close-ups of deposited particles are presented in* ***(B), (D)*** *and* ***(F)*** *The SEM images were provided by Philipp Zuber from the RUB.*

3.5.3 Fracture aperture and roughness

A modified version of the Fracture Trace Point (FTP) algorithm, developed and modified by Cappuccio et al. (2020), was used to measure local mechanical fracture apertures and the roughness of the segmented fractures in this study (see Figure 47 - 49 in the appendix). The algorithm is applied on a stack of 2-D binary images (with pixels in the background or matrix that are assigned a value of 0 and pixels in the fracture that are set to a value of 1, Figure 14A, E). In the first step, the 2-D Euclidean distance transform of all non-zero pixels is calculated:

$$px_i = \sqrt{\sum_{i=1}^{n}(px_i - b_i)^2} \tag{23}$$

with the number of dimensions (n), the position (in the corresponding dimension) of the fracture pixel (px_i) and its nearest background pixel (b_i). Each fracture pixel is assigned a value according to its distance to the nearest background pixel, i.e., pixels in the middle of the fracture are assigned higher values than pixels at the boundary to the background. Profiles across the fracture result in a peak-shaped distribution of the measured pixel values. Since the FTP-algorithm requires a valley-shaped distribution, the calculated distance values are inverted (Figure 14B, F). Then, in each image, profiles across the fracture are created orthogonal to the X- and Y-axis (not orthogonal to the fracture). From each profile, the FTP is determined, and the local apparent fracture aperture is measured as the distance from both grey value peaks (corresponding to the pixel values at the boundary between fracture and background) before and after the valley of the FTP (corresponding to the middle of the fracture). The apparent aperture values are corrected by considering the 2-D angle α and 3-D angle β (Figure 14C, G). Skeletonized 3-D images containing all local FTPs with their true aperture value are created (Figure 14D, H). The fracture roughness values are calculated, according to Zimmerman et al. (1991), by dividing the standard deviation of the measured aperture values by the mean aperture. Smoother fracture surfaces result in lower roughness values (i.e., two perfect, parallel, planar planes have a roughness value of 0).

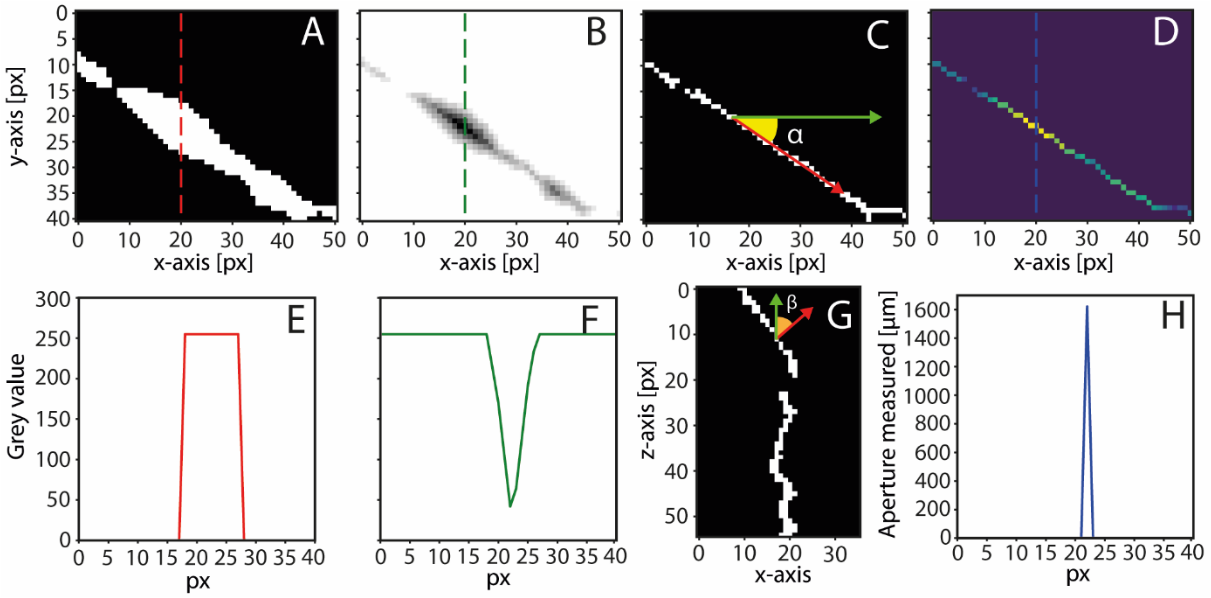


Figure 14. Workflow to calculate the true fracture apertures from a binary image stack
Profiles are illustrated as dotted lines and plotted below the corresponding image. The binary (segmented) images ***(A)*** *show a peak-shaped signal when measuring the grey values in profiles crossing the fracture* ***(E)****. The binary images are transformed by using the 2-D Euclidean distance transform, which is inverted* ***(B)*** *to obtain a valley-shaped profile* ***(F)****. The 2-D* ***(C)*** *and 3-D* ***(G)*** *local orientations are used to calculate the true aperture from the apparent aperture measurement.* ***(D)*** *The skeletonized fracture contains the true aperture values of the fracture. Reinhardt et al. 2022.*

Since the experimental data only provided the hydraulic aperture, the calculated mean mechanical aperture values were converted to hydraulic apertures to allow a meaningful comparison. As discussed in section 2.1, this conversion is not straightforward and depends on various fracture (e.g., roughness, contact areas) properties and hydraulic (e.g., flow velocity, Reynolds number) parameters. So far, there is no universally accepted conversion formula; thus, several formulas using different approaches exist (Souley et al. 2015; Kling et al. 2017). This study focuses on approximations of the hydraulic aperture (a_h), using the mean mechanical aperture (a_m) and its standard deviation (σ_{a_m}). Thereby, the relative roughness (see section 2.1 and subsections 3.5.3 and 4.1.3; calculated by dividing the standard deviation by the mean mechanical aperture) is considered in these approximations. Alternatives to this roughness coefficient are discussed in section 2.1 and include correlations based on other parameters, like the joint roughness coefficient or the Z2-value (Barton et al. 1985; Zhao et al. 2014). Table 2 presents the formulas applied to estimate the hydraulic aperture with the mean mechanical aperture and the standard deviation of the mechanical aperture.

Table 2. Selected equations for approximating the hydraulic aperture of the DTs
The hydraulic aperture (a_h) was estimated with the mean mechanical aperture (a_m) and standard deviation of the mechanical aperture (σ_{a_m}). Adapted from Kling et al. (2017).

No.	**Equation**	**Fracture type used for the derivation**	**Reference**
24	$a_h \approx \frac{a_m}{\sqrt[3]{(1+\frac{\sigma_{a_m}^2}{a_m^2})^{1.5}}}$	Theoretical equation based on log-normal aperture distributions	Renshaw (1995)
25	$a_h \approx \frac{a_m}{\sqrt[3]{1+20.5*(\frac{\sigma_{a_m}}{2a_m})^{1.5}}}$	Natural granite fractures	Barton and de Quadros (1997)
26	$a_h \approx a_m * \sqrt[3]{1-\frac{1.13}{1+0.191(\frac{2a_m}{\sigma_{a_m}})^{1.93}}}$	Tensile granite fracture	Matsuki et al. (1999)
27	$a_h \approx a_m * \sqrt[3]{1-\frac{\sigma_{a_m}}{a_m}}$	Replications of naturally fractured granite and split sandstone	Xiong et al. (2011)
28	$a_h \approx a_m * \sqrt[3]{0.94-5.0\frac{\sigma_{a_m}^2}{a_m^2}}$	Tensile sandstone fracture	Xie et al. (2015)
29	$a_h \approx \frac{a_m}{\sqrt{(1+\frac{\sigma_{a_m}}{a_m})^3}}$	Fracture sealed by compact quartz growth	Kling et al. (2017)

3.6 Simulation setups

Flow fields and permeabilities were calculated for the digitized fractures. These calculations were realized with the Left-Identity-Right (LIR) solver, implemented in the FlowDict module of GeoDict2020 (Linden et al. 2015). The LIR solver solves the partial differential equations (see section 2.5) to compute the force density field for incompressible Newtonian fluids. A non-uniform adaptive grid is applied to lower memory requirements. This approach performs efficiently in high porosity geometries (Linden et al. 2015). The permeability of the medium is computed by inputting the force density field into Darcy's law. Since a steady flow regime is assumed, the FlowDict module does not consider time-dependent phenomena (e.g., turbulence). However, the experimental conditions were adjusted to ensure the laminar flow condition (i.e., low Reynolds number), which assures the validity of the simulations. The LIR solver is capable of solving the Stokes equations (for linear flows) and the Navier-Stokes equation (non-linear flow, i.e., non-linear relation of pressure drop and mean velocity). For section 4.1, the Navier-Stokes equation was solved to compute flow fields and the Stokes equation for calculating the permeability. The permeabilities were computed by dividing the permeability over the whole simulated domain (the output of the simulations) by the fraction of fracture voxels in the model.

For the permeability computations only the digitized fractured core (+ background) was considered (without adding the geometries of the flooding cell, see Figure 15). The flow field and permeability computations (used for the tracer and barite particle transport simulations in sections 4.2 and 4.3) were realized by solving the Stokes equation since the calculated flows were in the linear regime (with mean velocities in the range of about 10^{-4} – 10^{-5} m/s and small Reynolds numbers, Schwarz and Enzmann 2013). Furthermore, the computational demands of solving the Stokes equation are lower than that of the Navier-Stokes equation. This was important for the calculations in section 4.2 since the voxel number (in each direction) of the DTs increased by a factor of 4 due to the rescaling to a resolution of 50 µm. The matrix (and background) was considered as impermeable quartz voxels (see section 3.1), and the fracture as permeable pore voxels.

In order to accurately compute the flow fields (and the tracer and particle transport in sections 4.2 and 4.3), the model was extended to mirror the experimental setup with the flooding cell and part of the tubes (Figure 15). The inflow and outflow tubing were added to the model as GAD objects (GeoDict analytic data), with a diameter equal to the diameter of the real tubing and a length equal to the distance between the core and tracer injection (inflow tubing) and the EC measurements (outflow tubing). The inflow and outflow areas between core and tubing were also added as GAD objects with the same diameter as the cores. Since each sample was positioned differently in the medical CT scanner, the inflow and outflow areas were created individually to ensure that each digital core was adequately connected to the inflow and outflow. The voxels of the GAD objects were assigned to the pore phase. The GadGeo module (implemented in GeoDict2020) was used to create the GAD objects (Streit et al. 2020).

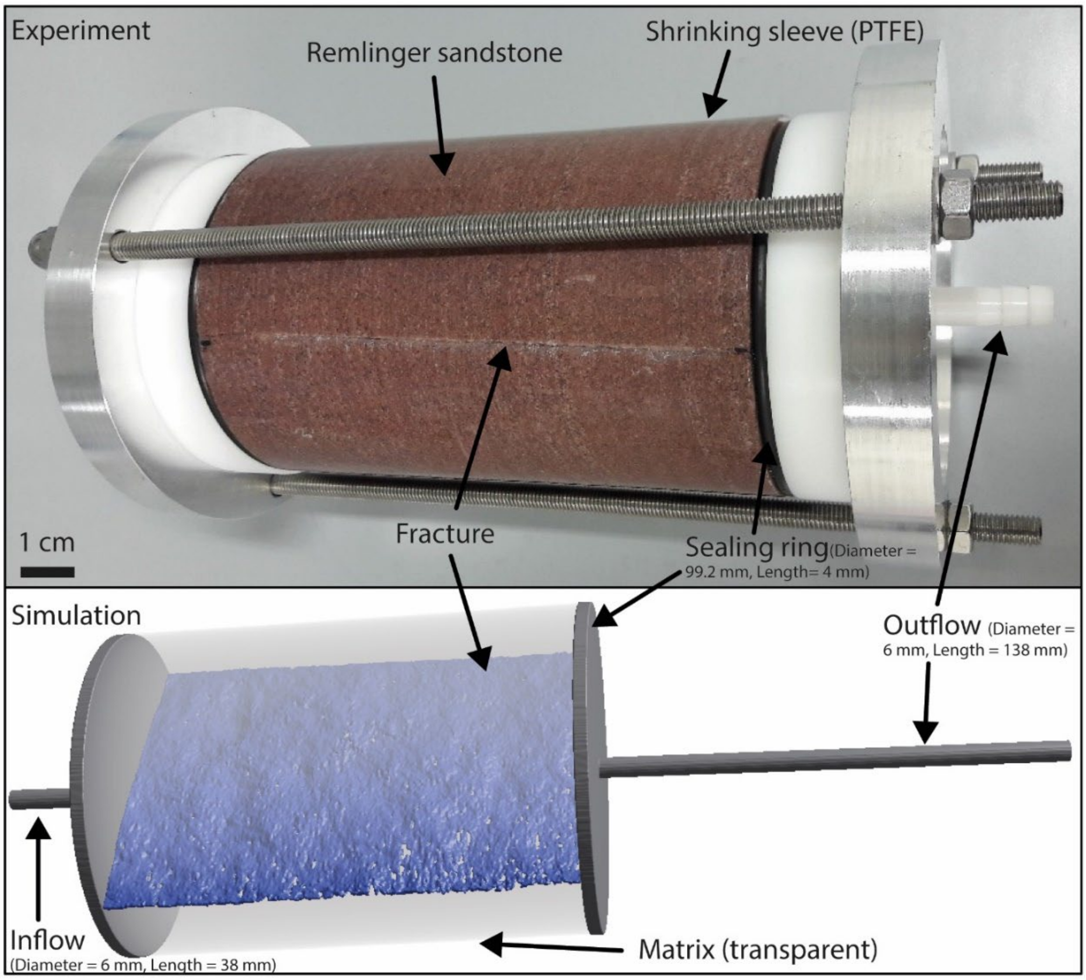

Experiment
Remlinger sandstone
Shrinking sleeve (PTFE)
1 cm
Fracture
Sealing ring(Diameter = 99.2 mm, Length= 4 mm)
Simulation
Outflow (Diameter = 6 mm, Length = 138 mm)
Inflow
(Diameter = 6 mm, Length = 38 mm)
Matrix (transparent)

Figure 15. Comparison of the experimental flow cell and the model
A Remlinger sandstone core was built into a flooding cell (top) to conduct water injection, tracer, and particle migration experiments. The experimental setup was translated into the model (bottom): the digitized core (fracture in blue, matrix in transparent grey) is joined by an inflow and outflow area, having the same size as the area enclosed by the sealing rings in the flooding cell. The length of the inflow and outflow pipes correspond to the distance between the core and tracer injection (inflow) and the location of EC measurement (outflow). Reinhardt et al. 2022.

In order to compare the segmentations for the Fle_8_S1 and Rem_1_S2 samples, the 500 preferred flow paths were calculated with the PoroDict module in GeoDict 2020. The algorithm identifies different paths to fit spheres through the structure, beginning with the biggest possible diameter (the maximum diameter option was used). The sphere diameter is iteratively reduced until 500 different paths through the structure are found. The symmetric boundary conditions were used for the calculations (Becker et al. 2020).

The modelling parameters for the structures in 200 µm resolution were chosen as follows. The fracture permeabilities and flow fields were computed with periodic boundary conditions (with an implicit region of 10 voxels at the inflow and outflow of the domain to avoid closing of flow channels during the simulation, see Hilden et al. 2020) in the flow directions and no-slip boundary conditions in tangential directions. The following parameters were applied to match the experimental conditions: temperature of 20°C, a slip length of 0 m (no sliding at the interface of solid and fluid), an error bound of 5 % as a convergence stopping criterion (i.e., the solver stops if the difference in predicted permeability is smaller than 5 %, within the last 100 iterations), a relaxation of 1, enabled grid refinement, LIR-Tree as the grid type, enabled multigrid method, parallelization on 48 cores and optimization for speed. For calculating permeabilities (by solving the Stokes equation) and flow fields of the fractures, a constant flow rate (0.3162 mL/s, measured for the Rem 1 S2 in the laboratory experiments) was applied over an area of 0.012996 m² (corresponds to the domain size of 570 x 570 voxels (X x Y direction), with a resolution of 200 µm). These settings resulted in similar flow velocities as determined in the experiments and were therefore applied for both samples.

For the reconstructed structures in 50 µm resolution, the same parameters were applied for the calculation of the flow fields. In order to achieve a modelling setup closer to the real experimental condition, a constant pressure was applied for each simulation with the ten fractured samples. The computed flow fields were used to simulate the tracer experiments with the AddiDict module (implemented in GeoDict 2020). Infinitesimal small particles (= molecules) were used as conservative tracer particles with the sieving collision model. The restitution value was set to 1 (a particle did not lose energy when colliding with solid materials). The advective and diffusive motion of the tracer particles was considered. The diffusion was considered by setting the diffusion coefficient of water at 20°C (2.02×10^{-9} m²/s, Holz et al. 2000) as the diffusion coefficient of the tracer particles. The other parameters were set as follows: a simulation time of 3000 s, a time step of 1 s (the same as the measuring interval in the experiment), periodic BC in the tangential directions, open BC at the upper end of the domain (i.e., particles are only allowed to leave the domain when reaching the outflow. Only escaped particles are considered in the computed breakthrough curves), reflective BC at the lower end of the domain, 1×10^5 particles were created simultaneously in the first ten slides (in Z-direction) of the fractures. The number of particles was chosen due to computational limitations. The starting velocity of the particles equalled the flow velocity in the corresponding voxel. The molecular particle motion (dominated by advection) was calculated according to equation 11 (Weber et al. 2020). Details regarding the choice of the modelling settings of the reconstructed fractures can be found in subsection 4.2.1.

In order to reproduce the barite particle migration experiments, the settings described above were adjusted to the experimental conditions in the HPTC. For flow field calculations, the temperature was set to 58°C, and a constant pressure was applied to produce flow rates between 1 – 2.0323 mL/s (depending on the individual settings for each experiment, more details can be found in subsection 3.5.2). The setup was flipped to position the inflow at the top of the domain and the outflow at the bottom of the domain. The flow was applied from top to bottom with a negative pressure drop (the flow in the experiments was applied in the same way). The following parameters were changed for the particle migration simulations: the simulation time was reduced to 500 s, 3×10^6 particles were created in the first ten slides (in Z-direction), the density of the barite particles was set to 4480 kg/m³ (Johnson et al. 2017), the Hamaker collision model with a Hamaker constant of 10^{-23} J (more details regarding the choice of these parameters can be found in section 4.3). The particle movement (Brownian diffusion and advection) was computed by solving the equations presented in section 2.6. In order to analyse the influence of each parameter on the barite deposition, a sensitivity study was conducted with the following parameters and value ranges: particle sizes between 1 – 42 µm, a particle amount of 0.5 – 6 $\times 10^6$, Hamaker constants of 10^{-22} – 10^{-24} J, temperatures between 38 – 78°C and flow rates of 0.5 – 8 mL/s. Additionally, the random seed was varied to estimate a numerical error of the simulations (the parameters are also summarized in Table 10 in subsection 4.3.2).

4. Case studies

This chapter presents the key results of the three case studies conducted for this thesis. The first section focuses on the comparison of different segmentation techniques, i.e., conventional (several thresholding methods, watershed) with more recently developed methods (RF, CNN). The segmentation, obtained by applying the most suitable segmentation method, is the basis for the reconstruction workflow described in section 4.2. Section 4.2 also presents the validation of the finer resolved (50 µm) fracture models with experimental data, and the comparison with the 200 µm resolved fracture models. In section 4.3, the final fracture models were used to reproduce both successful barite migration experiments. Finally, a sensitivity study was conducted to analyse how the barite deposition pattern is influenced by various parameters. Each case study is concluded with a discussion of the results and an outlook for future studies.

4.1 Comparison of fracture segmentation methods

4.1.1 Conventional segmentation methods

In order to determine the most suitable segmentation method for the presented samples, two representative samples (Fle_8_S1 and Rem_1_S2) were chosen to quantify the quality of the output received by applying various methods. The samples were segmented by using various thresholding methods and the watershed technique. The reconstructed medical XCT-images were pre-processed and masked as explained in section 3.2. Since all grey value histograms (Figure 6) show a complex distribution without distinct peaks for the two phases (fracture and matrix), the segmentation was expected to be challenging for conventional methods (Al-Marzouqi 2018).

Figure 16 and Figure 17 present one exemplary slide from both samples to visually compare each segmentation result. Figure 16 shows the evolution of the segmentation for the Flechtinger sample when the threshold is progressively raised. Each segmentation result for the Remlinger sample can be seen in Figure 20 and for the Flechtinger sample in Figure 19. Similar results were observed for both samples: A low maximum threshold (Fle_8_S1: 3360 – 3380, Rem_1_S2: 3350 – 3380) failed to segment large portions of the fracture, especially near the edges of the core. On the other hand, with higher maximum thresholds (Fle_8_S1: 3400 – 3490, Rem_1_S2: 3400 – 3440), more fracture area was segmented, but there were still parts of the fracture that were not correctly segmented. Furthermore, a higher maximum threshold led to considerably more matrix voxels that were falsely segmented as fracture voxels (in comparison to lower maximum thresholds). These errors were removed by applying the "cleanse" tool implemented in GeoDict 2020. However, with higher thresholds (Fle_8_S1: starting at 3400, Rem_1_S2: starting at 3420), these errors merged with the fracture. This altered the true fracture surface, i.e., local apertures and the roughness values were differing more and more from the real fracture. These errors cannot be corrected after applying the segmentation. Therefore, a trade-off is observed: a higher maximum threshold segmented more fracture area but introduced more errors at a lower maximum threshold. The amount of erroneously segmented matrix voxels can be seen in Figure 17 B-C. The threshold segmentation at a value of 3420 led to a porosity (i.e., fracture fraction) of 1.81 %, after applying the "cleanse" tool (i.e., removing the errors in the matrix), the porosity dropped to 1.18 %, a difference of about 35 %.

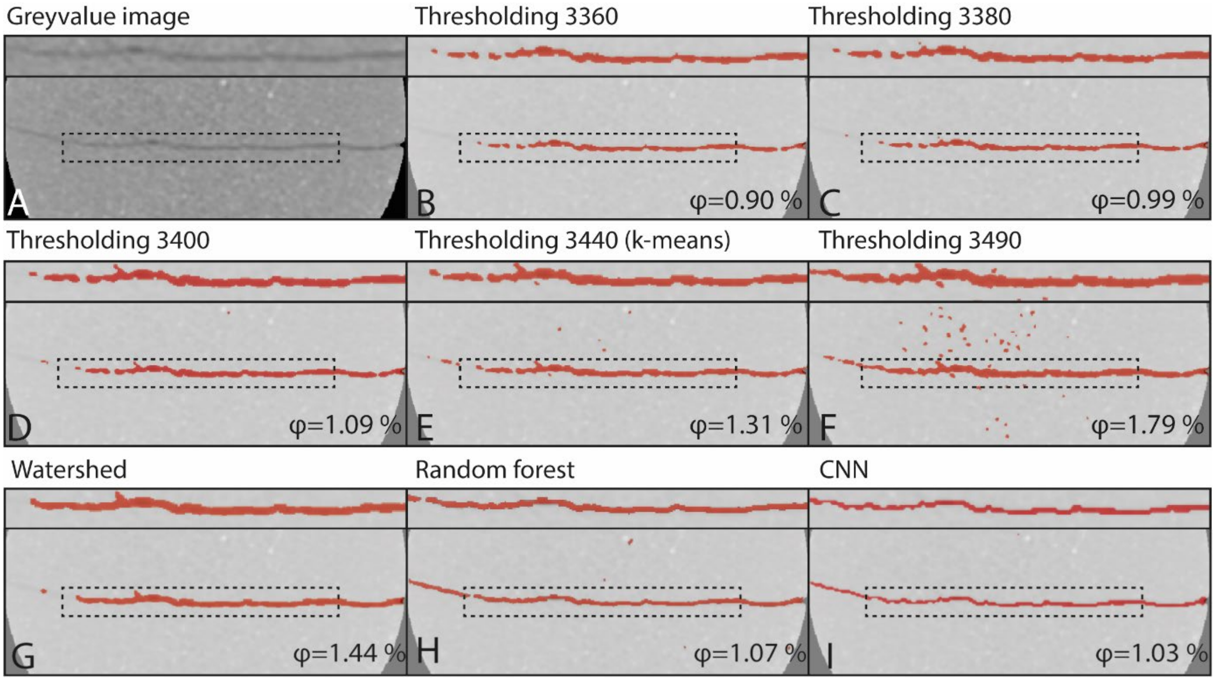


Figure 16. Comparison of the applied segmentation methods on the Fle_8_S1
Each window presents a close-up of the middle fracture zone. ***(A)*** *Slice 519 of the masked pre-processed greyscale medical XCT-image stack. Segmentation results using thresholds with the ranges: 10 – 3360* ***(B)****, – 3380* ***(C)*** *– 3400* ***(D)*** *– 3440 (****E****, obtained by the K-means clustering), – 3490* ***(F)****.* ***(G)*** *The segmentation after applying the watershed method. The fracture is marked in red.* ***(H)*** *Segmentation obtained with the RF classifier of the machine learning software Ilastik (Version 1.3.3).* ***(I)*** *Segmentation obtained with the CNN. The RF classifier and CNN were not trained on this slide. For each segmentation, the porosity φ is presented.*

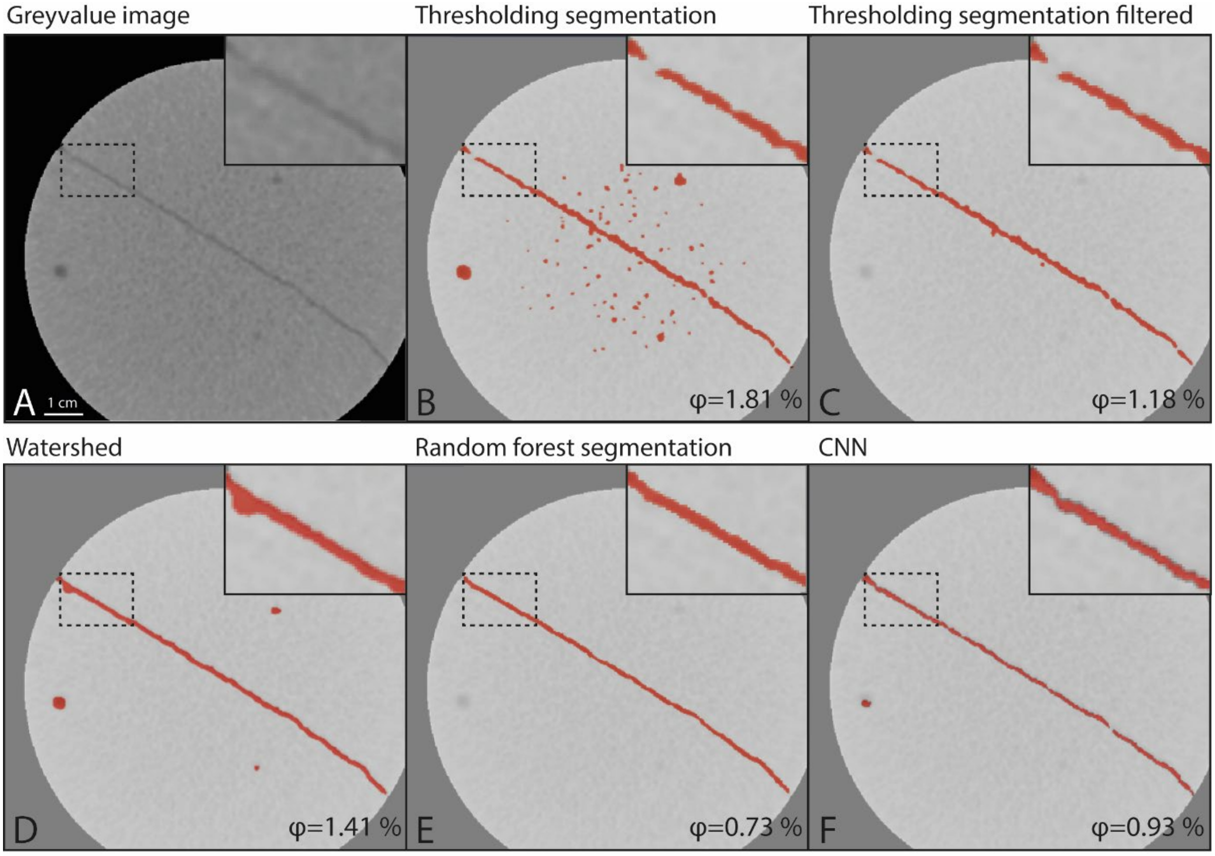

Greyvalue image
Thresholding segmentation
Thresholding segmentation filtered
A
1 cm
B
φ=1.81 %
C
φ=1.18 %
Watershed
Random forest segmentation
CNN
D
φ=1.41 %
E
φ=0.73 %
F
φ=0.93 %

Figure 17. Comparison of the applied segmentation methods on the Rem_1_S2
***(A)** Slice 130 of the masked pre-processed greyscale medical XCT-image stack. **(B)** The segmented fracture (red), using a global threshold with the range 10 – 3420. **(C)** The result after applying the "cleanse" tool (implanted in GeoDict 2020) on the segmentation result (shown in (B)). Segmentation results obtained by using the watershed algorithm **(D)**, RF classifier **(E)**, and CNN **(F)**. The RF classifier and CNN were not trained on this slide. For each segmentation, the porosity φ is presented. Adapted after Reinhardt et al. 2022.*

In order to include a reasonable selection of segmentations, thresholds were applied to compare partially segmented fractures without errors in the matrix (Fle_8_S1: 3360, Rem_1_S2: 3350) up to widespread segmented fractures with high errors (Fle_8_S1: 3490, Rem_1_S2: 3440). For the Fle_8_S1, the global threshold computed by the K-means clustering method (3440) lay in the middle of the considered threshold ranges. However, the global threshold computed by the Otsu method (Fle_8_S1: 3662) and both calculated thresholds for the Rem_1_S2 (Otsu: 3582, K-means clustering: 3588) were not considered in the further analysis since these thresholds led to extensive errors in the segmentations. Numerous matrix voxels were falsely classified as fracture voxels and merged with the fracture, which prevented meaningful segmentation results. The threshold methods were not able to deal with the ring artefacts (Figure 7D). The resulting errors were removed by cropping the datasets in the Z-direction.

Visually, the watershed algorithm performed similarly to the thresholding methods. Much of the fracture area was correctly segmented, and only a few islands of falsely segmented matrix voxels were observed. However, this method showed some problems in distinguishing the border between the fracture and matrix. Matrix voxels with a similar grey value as the fracture voxels led to a low gradient between the phases and were falsely segmented as fracture voxels (Figure 16G, Figure 17D). Like the thresholding methods, the watershed algorithm failed to segment large parts of the fracture, especially in the outer regions of the cores. The ring artefacts were falsely segmented by the watershed algorithm and had to be removed by cropping the datasets in Z-direction.

4.1.2 Machine learning methods

In order to obtain the most robust segmentation, the masked and pre-processed XCT-images were segmented with the RF classifier using all predefined filters: Gaussian Gradient Magnitude, Laplacian of Gaussian, Gaussian Smoothing, Difference of Gaussians, Structure Tensor Eigenvalues and Hessian of Gaussian Eigenvalues (see subsection 2.4.2). Each filter was applied with every predefined sigma value (of 0.3 (only for the Gaussian Smoothing), 0.7, 1.0, 1.6, 3.5, 5.0, and 10.0). The training data was provided manually with brush strokes. Preliminary results and uncertainty maps were used to refine the training data and minimize segmentation errors. For the final segmentation result, 0.0285 % (for the Rem_1_S2) and 0.0264 % (for the Fle_8_S1) of the voxels were used as training data (Table 3). The voxels were segmented as “background”, “fracture”, or “matrix” phase.

Table 3. Amount of training data provided to the RF classifier
Both datasets consist of 256,346,100 voxels, which were segmented as background, fracture, or matrix phase.

Sample	Trained matrix voxels	Trained fracture voxels	Trained background voxels
Fle_8_S1	63,095	4,515	5,481
Rem_1_S2	58,806	1,905	6,849

Although only a limited amount of training data was provided to the RF classifier (see Table 1), a visual comparison revealed that it provided superior segmentation results compared to the results obtained by the conventional methods. For both representative samples, the whole fracture was segmented, and only a few matrix voxels were falsely segmented as fracture voxels (Figure 16H, Figure 17E). These errors were removed with the “cleanse” tool of GeoDict 2020. Interruptions in the fracture area were visually interpreted as locations with low fracture aperture (too small to be detected in the 200 µm resolved grey value medical XCT-images) or contact areas between the opposing fracture surfaces (Figure 18A, B). Moreover, during the training phase, the constant interaction between the user and the RF classifier (providing preliminary segmentation results and uncertainty maps) reduced user bias and allowed for a more precise training. An uncertainty map is illustrated in Figure 18. The highest uncertainties (red – yellow colours) mainly occurred at the boundary between the fracture and the matrix because the PVE causes a blurred fracture area. This effect makes distinguishing between fracture and matrix challenging. To ensure a meaningful segmentation result, it is, therefore, crucial to provide proper training data to the RF classifier since the boundary voxels highly influence the fracture properties (aperture and roughness).

The RF classifier was able to handle noise and the ring artefacts (Figure 18D). Therefore, the segmented datasets did not need to be cropped as extensively as the segmentations obtained by the conventional methods. This allowed the creation of a more realistic 3-D model. However, to equally compare the results, the RF segmentations were cropped in the same way as the results from the conventional methods.

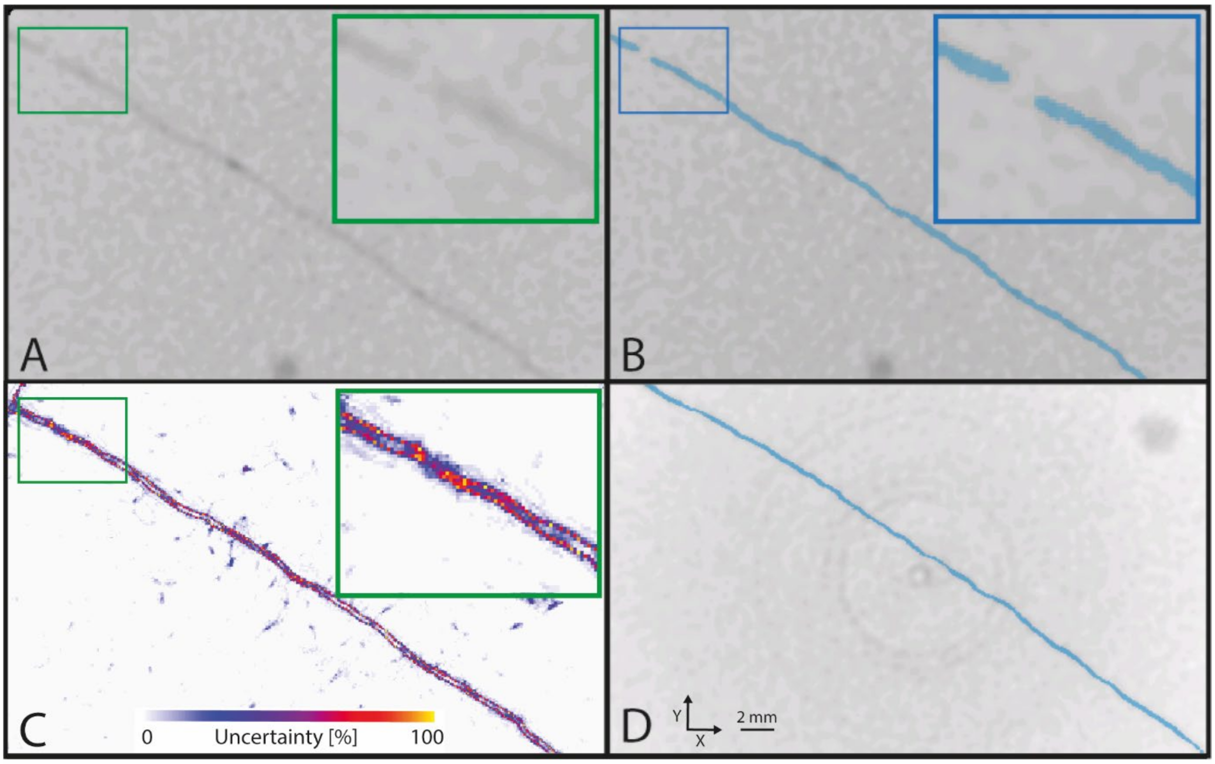


Figure 18. Results of the RF segmentation
Slice 273 of the masked pre-processed greyscale medical XCT-image stack of the Rem_1_S2 sample ***(A)****.* ***(B)*** *Segmented image. The fracture is highlighted in blue; the greyscale image is shown in the background.* ***(C)*** *The uncertainty map of the segmentation result shown in (B).* ***(D)*** *Segmented fracture with the greyscale image of slide 760 shown in the background. The RF classifier correctly segmented the fracture despite the ring artefacts. The RF classifier was not trained on the presented slices. Adapted after Reinhardt et al. 2022.*

As the RF classifier, the CNN provided a superior segmentation for both samples when compared visually (Figure 16I, Figure 17F). The whole fracture area was well segmented, uninfluenced by the artefacts. Few errors occurred in the matrix but were removed by applying the “cleanse” tool of GeoDict 2020. Table 4 presents the final accuracy and loss functions for both training processes.

Table 4. Loss and accuracy values of the CNN segmentation approach
In Training I, the CNN was trained on the Rem_1_S2 sample, and the validation was applied on the Fle_8_S1 sample. In Training II, the training and validation datasets of Training I was switched.

	Training I		**Training II**	
	Training	**Validation**	**Training**	**Validation**
Loss	0.0043	0.0037	0.0030	0.0032
Accuracy	0.9975	0.8326	0.9982	0.8873

4.1.3 Fracture properties

Several fracture properties were calculated for each fracture. Table 5 summarizes the results of the porosity (fracture portion), permeability, mean mechanical aperture, and roughness calculations. Depending on the applied segmentation method, these properties showed distinct deviations from each other. However, similar results were produced by different segmentation methods. For instance, in case of the Fle_8_S1, similar permeability and porosity values were calculated for the fractures obtained by the RF classifier and the thresholding at 3380 and 3400. Since a visual comparison (Figure 16C, D, H) revealed considerable differences in the segmentation results (and their quality), the calculated parameters were not sufficient to quantify the quality of the different segmentation methods and did not allow choosing the most suitable method. For a better comparison, the flow fields and preferred flow paths were calculated and are presented in the next subsection.

Table 5. Overview of segmented fracture properties
Results of the permeability, porosity, roughness and mean mechanical aperture calculations for each segmented fracture of the Fle_8_S1 and Rem_1_S2 sample.

	Segmentation method	**Permeability x10^{-8} [m^2]**	**Porosity (fracture portion) [%]**	**Roughness [-]**	**Mean mechanical aperture [µm]**	**STD [µm]**
Fle_8_S1	**Thresholding at 3360**	3.32	0.90	0.38	885	331
	Thresholding at 3380	4.15	0.99	0.36	935	336
	Thresholding at 3400	5.05	1.09	0.35	985	343

	Thresholding at 3410	5.45	1.15	0.34	1003	339
	Thresholding at 3430	6.55	1.26	0.33	1017	354
	Thresholding at 3440 (K-means clustering)	7.07	1.31	0.33	1065	361
	Thresholding at 3450	7.48	1.39	0.37	1082	399
	Thresholding at 3470	8.41	1.55	0.43	1089	464
	Thresholding at 3490	8.96	1.79	0.57	1092	576
	Watershed	10.20	1.44	0.22	1201	269
	RF classifier	4.67	1.07	0.22	827	179
	CNN	4.19	1.03	0.22	816	177
Rem_1_S2	**Thresholding at 3350**	0.65	0.53	0.41	692	286
	Thresholding at 3380	1.92	0.78	0.39	805	314
	Thresholding at 3400	3.11	0.98	0.39	887	346
	Thresholding at 3410	3.79	1.08	0.39	929	370
	Thresholding at 3420	4.53	1.18	0.41	973	396
	Thresholding at 3430	5.28	1.31	0.44	1009	443
	Thresholding at 3440	6.00	1.46	0.49	1032	513
	Watershed	6.91	1.41	0.76	1248	952
	RF classifier	2.24	0.73	0.22	571	126
	CNN	2.98	0.93	0.26	787	206

Fle_8_S1

Figure 19 presents the segmentation result after applying the conventional and machine learning segmentation methods. For each structure, the flow fields and 500 preferred flow paths were calculated and are illustrated in Figure 19. As mentioned in subsection 4.1.1, a trade-off was observed for the thresholding methods. The segmented structures obtained from applying a low threshold did not correctly represent the margins of the fracture (in X-direction). Gradually more area of the margins was segmented when applying higher thresholds; however, this also led to an increased number of errors. For example, for the thresholding segmentation with values of 3470 and 3490, many matrix voxels near the inflow were incorrectly segmented as fracture voxels, leading to significantly bigger fracture apertures at the inflow (left side of the structures in Figure 19). The flow velocities were increasing with higher thresholds, and the flow was more evenly spread. Since the upper margin of all threshold structures (in X-direction) was not correctly segmented, none of the calculated 500 preferred flow paths was located in this area. The watershed segmentation performed like a high threshold segmentation: higher flow velocities and evenly spread flow paths (except for the upper margin in the X-direction) were observed. The segmentation errors at the inflow were not observed, i.e., visually, the watershed segmentation provided a slightly better result than the thresholding techniques. Both machine learning approaches segmented the whole fracture area, resulting in evenly distributed flow paths and flow fields. No obvious segmentation errors could be observed, which qualitatively indicated that the structure was better represented by the machine learning segmentations.

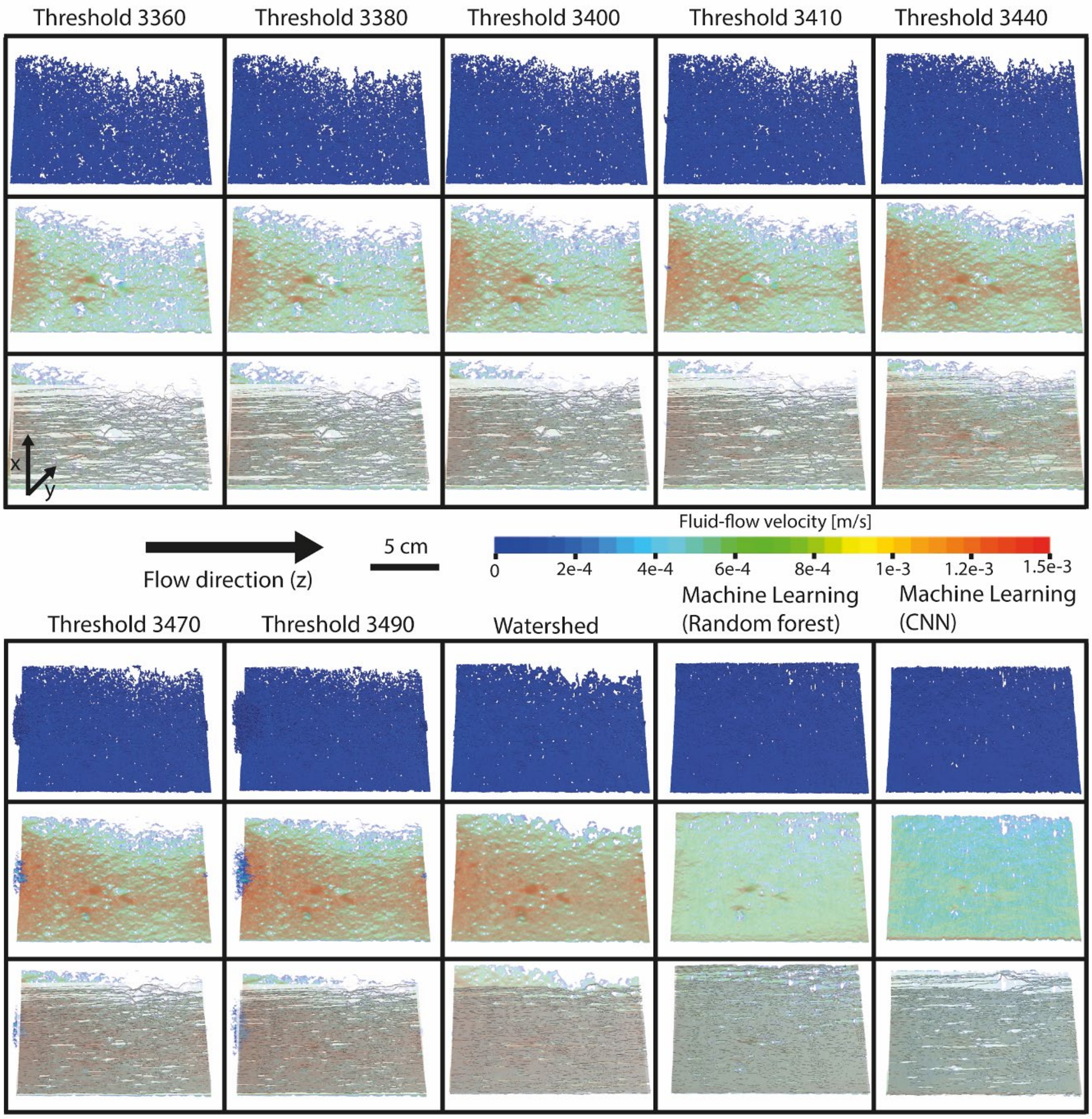


Figure 19. Comparison of segmented structures of the Fle_8_S1
Comparison of the fractures (blue, upper panels), calculated flow fields (middle panels), and 500 preferred flow paths (bottom panels) for each structure, obtained by applying various segmentation methods on the Fle_8_S1 sample. Inflow and outflow areas are not shown. Adapted after Reinhardt et al. 2022.

Rem_1_S2

Figure 20 presents the comparison of each segmented structure on the basis of the fracture area (in blue), the flow field, and the calculated 500 preferred flow paths. The outer margins (in X-direction) were not fully segmented by the thresholding techniques. Higher thresholds led to more segmented fracture areas, but even with high threshold values, the margins were not correctly segmented. This was also observed in the flow fields and the preferred flow paths, which were concentrated in the middle. The watershed segmentation showed high flow velocities and flow paths concentrated in the middle of the fracture. This was caused by large areas near the margins in the X-direction, which were not segmented. The RF classifier segmented the whole fracture area, which led to flow paths evenly distributed across the fracture. Also, the CNN segmentation resulted in an evenly segmented fracture. However, more contact areas between the fracture walls (i.e., matrix voxels disrupting the fracture area) were present. The flow velocities were higher than in the RF segmented fracture, and the flow paths were more concentrated.

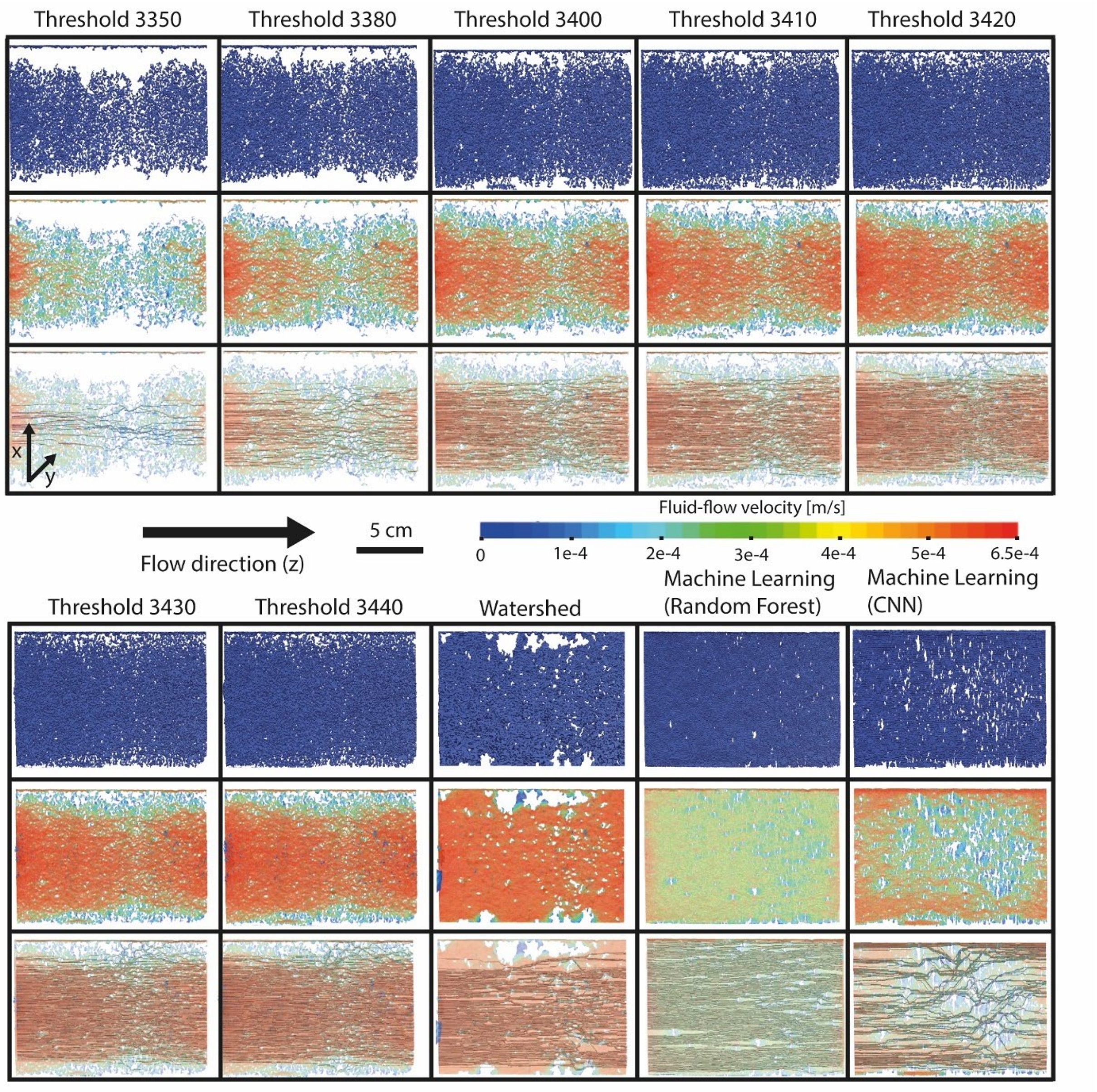


Figure 20. Comparison of segmented structures of the Rem_1_S2
Comparison of the fractures (blue, upper panels), calculated flow fields (middle panels), and 500 preferred flow paths (bottom panels) for each structure, obtained by applying various segmentation methods on the Rem_1_S2 sample. Inflow and outflow areas are not shown. Adapted after Reinhardt et al. 2022.

Since permeability, porosity, and roughness values overlapped between the different results, these parameters did not allow for a quantitative comparison of the structures obtained by applying different segmentation techniques. However, the visual comparison of the 2-D structures, flow fields, and flow paths showed large deviations and allowed a qualitative evaluation of the methods.

4.1.4 Discussion and conclusion

In this section, the benefits of machine learning segmentation methods were demonstrated on two exemplary samples (Fle_8_S1 and Rem_1_S2). Due to the complex grey value distributions (Figure 6) of the samples, the segmentation was expected to be challenging for the conventional segmentation approaches. This was confirmed by comparing flow fields, preferred flow paths, and the visualization of the segmented structures (in 2-D and 3-D). Various errors were observed for the conventional methods (thresholding, watershed segmentation): large areas of the fractures were falsely segmented as matrix voxels (especially at the margins in the X-direction), matrix voxels were falsely segmented as fracture voxels (due to similar grey values), and some merged with the fracture structure (the extend depended on the applied threshold) and ring artefacts severely influenced the segmentation in the upper and lower slices (in the Z-direction). These errors highly influenced the fracture properties (apertures, roughness, porosity) and the calculated flow fields. After providing appropriate training data, both machine learning approaches (RF and CNN) performed significantly better. The influence of the ring artefacts was negligible, and a larger structure could be extracted. However, for a proper comparison, the obtained structures were cropped in the same way as the structures from the conventional methods. The machine learning methods allowed the segmentation of the whole fracture area with minor intrusions of matrix voxels, which were interpreted as narrow apertures (not detectable by the medical XCT scanner) or contact areas between the opposing fracture sides. Few errors in the matrix were observed (islands of matrix voxels, segmented as fracture voxels, Figure 7D). Evenly distributed flow fields and flow paths were calculated.

Fracture properties (permeability, porosity, roughness, mean mechanical aperture) were dependent on the applied segmentation method and thus showed large deviations for the obtained DTs. However, these parameters were not suitable for the quantitative comparison since results obtained from different segmentation methods showed similar fracture properties, while a visual comparison revealed considerable differences in the structures. For example, in the case of Fle_8_S1, similar permeability and porosity values were calculated for the segmentation obtained by the RF classifier and two thresholds. A visual comparison showed obvious errors in the fractures obtained from the conventional approach.

Overall, the permeabilities of the segmented fractures deviated more than one order of magnitude from the experimental data (1.73×10^{-9} m² for the Flechtinger and 0.731×10^{-9} m² for the Remlinger fracture), and the mean fracture apertures (mechanical) of the DT were about five times higher as the measured apertures (hydraulic). These deviations could be explained by the influence of the PVE, which is more significant with smaller apertures, low XCT scan resolutions, and at structure boundaries (section 2.3). At a given spatial resolution, the extraction of a representative fracture model gets more challenging with lower fracture apertures (Ketcham et al. (2010). Since the calculated hydraulic aperture (Frank et al. 2020a) of the Rem_1_S2 (93.7 µm) is smaller than the hydraulic aperture of the Fle_8_S1 (143.9 µm), the PVE is expected to have a greater influence on the Remlinger sample. However, both mean apertures (hydraulic) are below the spatial resolution of the XCT-scans, and hence a considerable influence of the PVE on both samples is expected. This is reflected in the large deviations in permeability and mean aperture for both samples. The next section investigates how the PVE affects the digitization of the fractures, and a correction method is applied.

The machine learning approaches showed several advantages over the traditional segmentation methods. After appropriate training, the CNN method can efficiently handle image artefacts, low resolutions, and high noise-to-signal ratio. In contrast, traditional methods require image pre-processing when the acquired images suffer from these problems. This pre-processing is often realized by applying image filters, which could lead to a loss in image information (Iassonov et al. 2009; Saxena et al. 2017b; Lee et al. 2022) The disadvantages of the CNN approach include the selection of the optimal network architecture (different architectures should be tested) and parameters (e.g., learning rate, optimizer, number of filters, batches, and epochs), which are computationally demanding and time-consuming steps (Lee et al. 2022). Furthermore, a sufficient number of training data and an appropriate ground truth are the foundation of the CNN approach. It can be challenging, if not impossible, to get enough training data and an appropriate ground truth, especially when working with unique samples (Wang et al. 2021a, also see subsection 2.4.2).

Complex images and artefacts are also handled well with the RF approach (Berg et al. 2018). The method is user-friendly and accessible through several open-source software packages, which provide an adjustable, pre-defined workflow. A ground truth is not required since the training is provided by the user directly on the grey value images (i.e., supervised learning). For the RF implementation used in this study, an interactive training phase minimized user bias by providing preliminary results and uncertainty maps. Since the training in this approach is conducted on one 2-D image at a time, the computational demands of this method are low. The greatest disadvantage of the method is that user bias is minimized but not completely eliminated. The RF method has significant advantages over the other methods tested in this study, especially when working with unique samples. The comparison in this section highlighted the superior segmentation results obtained by the RF classifier. Due to the lack of other ground truth data (e.g., SEM images), the RF segmentation was used as a reference for the CNN segmentation method. Both approaches resulted in similar segmentations of the fractures, but the preferred flow paths and flow fields indicated a better performance of the RF segmentation. Therefore, this method was applied for segmenting each of the ten samples used in the further analysis.

The following section investigates the influence of the PVE by applying a novel fracture reconstruction workflow (also see section 3.4). As validation, the permeability and fracture volumes were compared with experimental data and hydraulic apertures were estimated from mechanical aperture calculations using various conversion formulas. The segmentations using the RF classifier were compared with the reconstructed fractures and the lab data to highlight the influence of the PVE and the difference in fracture properties. Additionally, the tracer experiments were simulated on the reconstructed fractures to verify the DT with additional parameters.

4.2 Fracture reconstruction

Various segmentation methods were compared, and the advantages of the RF approach were highlighted in section 4.1. Permeabilities and the mean mechanical apertures of the segmented fractures showed significant deviations from experimental data, which was attributed to the PVE. This section focuses on diminishing the influence of the PVE on the segmented fractures. For this, a novel reconstruction approach was applied to obtain better resolved fracture models. These structures were validated by experimental data (permeability, apertures, breakthrough curves) and compared to the segmented fractures.

This section is structured in the following way: The first subsection describes the workflow, results, and applied parameters to obtain the reconstructed fractures. The method is described in section 3.4 in detail. The next subsection presents the comparison with data from laboratory experiments. Exemplary, these steps were highlighted on the same samples used in the previous section (Fle_8_S1, Rem_1_S2). Afterwards, the reconstructed fractures were compared to the segmented binary images to highlight the difference between the 50 µm and 200 µm resolved fracture models. The final subsection summarizes the results and gives an outlook for future studies.

4.2.1 Workflow

Each medical XCT image stack was segmented by using the most appropriate segmentation method (i.e., the RF classifier). These binary images and the grey value images (resampled to a resolution of 200 µm, image type converted to 16-bit unsigned, with grey values raised by 2000) were used to reconstruct the fractures at a higher resolution.

Details regarding the reconstruction workflow can be found in section 3.4. CT_{air} values of 1011 – 1013 HU and global CT_{mat} values of 3672 – 3702 HU (Flechtinger samples) and 3575 – 3590 (Remlinger samples) were calculated. Since the grey values in the images were raised by 2000 HU (to eliminate negative values prior to the segmentation), the CT values must be shifted by 2000 HU to be comparable with literature data. A comparison with literature data revealed similar values (at similar scanning settings) for the CT values of air (-1000 HU is usually recommended as an approximation) and the sandstone matrix of various sandstone types with values in the range of 1100 – 1900 HU (Hounsfield 1980, Vinegar et al. 1991; Akin and Kovscek 2003; Huo et al. 2016; Kling et al. 2016).

The reconstructed fractures were validated by simulating a tracer breakthrough curve. The settings were discussed in section 3.6. The GAD objects added to the 200 µm resolved fractures (Figure 15) were resampled to a resolution of 50 µm. The diffusion coefficient of water at 20°C (2.02×10^{-9} m²/s) was set as the diffusion coefficient of the tracer particles (Holz et al. 2000). For computing the flow field, a constant flow rate of 0.3162 mL/s (measured in the experiments) over an area of 4.2 cm² was applied for the Rem 1 S2 sample (with a computed pressure drop of 159.51 Pa). The previous settings (with a flow rate of 0.3162 mL/s over an area of 0.012996 m² for the simulation of the flow fields in section 4.1) resulted in significantly slower flow velocities in the reconstructions since the settings were chosen to reproduce the measured flow velocities in the considerably larger 200 µm segmentations. Thus, the settings for the flow field simulation had to be corrected to replicate the flow conditions applied in the laboratory. The value for the area was chosen to match with the average flow area of the Rem 1 S2 core (consisting of the 2-D flow-area (in flow direction) of the fracture estimated in the laboratory experiments and the inflow/outflow areas in the simulation). This flow condition resulted in flow velocities in the range of 10^{-4} - 10^{-5} m/s, which were similar to the flow velocities measured in the experiments. Since the experiments were conducted with a constant pressure drop, the computed pressure drop in the Rem 1 S2 sample (159.51 Pa) was applied. Due to the limited resolution, small scale features (roughness, small apertures, tortuosity) could not be accurately represented in the reconstructed fractures. Since these features have a significant influence on the pressure drop across a fractured core (e.g., Mofakham et al. 2018), the calculated pressure drop from the simulation is underestimated and cannot be directly compared with the pressure drop applied in the experiments. The applied pressure drop for the simulations resulted in plausible flow velocities and was therefore also used for the simulations conducted with the other nine fractured samples. For the particle simulations, two additional adjustments are applied:

(i) The resampled structures were cropped by 200 voxels in the X+, X-, Y+, and Y- directions (to omit unnecessary background voxels from the model), 1000 voxels in the Z+ direction (outflow pipe) and 700 voxels in the Z- direction (inflow pipe).

(ii) The $1x10^5$ particles (limited by the computational resources) were created in the first ten voxels (in the Z-direction) of the fracture, with a minimum distance of 300 - 400 voxels to the model boundary in the X-direction.

Due to the resampling to the resolution of 50 µm, adjustment (i) had to be applied to lower the computational demands and make the simulations feasible. Due to high flow velocities in the inflow and outflow pipes, reducing the length of the pipes did not notably influence the breakthrough curves. Adjustment (ii) was applied because of two reasons. Firstly, when the particles were created in the inflow pipe (as realized in the simulations of subsection 4.1.4), only a few particles entered the fracture because of the small fracture opening (this effect was also observed in the barite migration experiments at the RUB). This led to a low number of particles detected at the outflow of the domain, which was not sufficient to plot a breakthrough curve. A significant increase in the number of particles to overcome this problem was computationally not feasible. Secondly, in test particle simulations on the 200 µm fractures, it was observed that the particles predominantly entered the fracture in the middle of the fractures. Since the exact inflow function was not captured in the laboratory experiments, as a simplifying assumption, the particles were created simultaneously.

The simulated breakthrough curves were compared to the measured breakthrough curves. Two parameters were adjusted during the reconstruction workflow until a match with the tracer data was acquired, and thus, a representative DT was created. Either local or global CT_{mat} values were applied, and the growing limit was adjusted. These parameters were varied to account for uncertainties in the reconstruction process (e.g., errors in the local aperture calculation or errors in the calculation of the CT values) and the limit of the resampled resolution (i.e., a resolution of 50 µm might still not be sufficient to adequately resolve the fractures). More details about the workflow can be found in section 3.4. A discussion of error sources in the reconstruction workflow and proposals for benchmarking studies can be found in subsections 4.2.2 and 4.2.4. Table 6 summarizes the parameter settings for each sample. Figure 21 presents an overview of the reconstruction workflow.

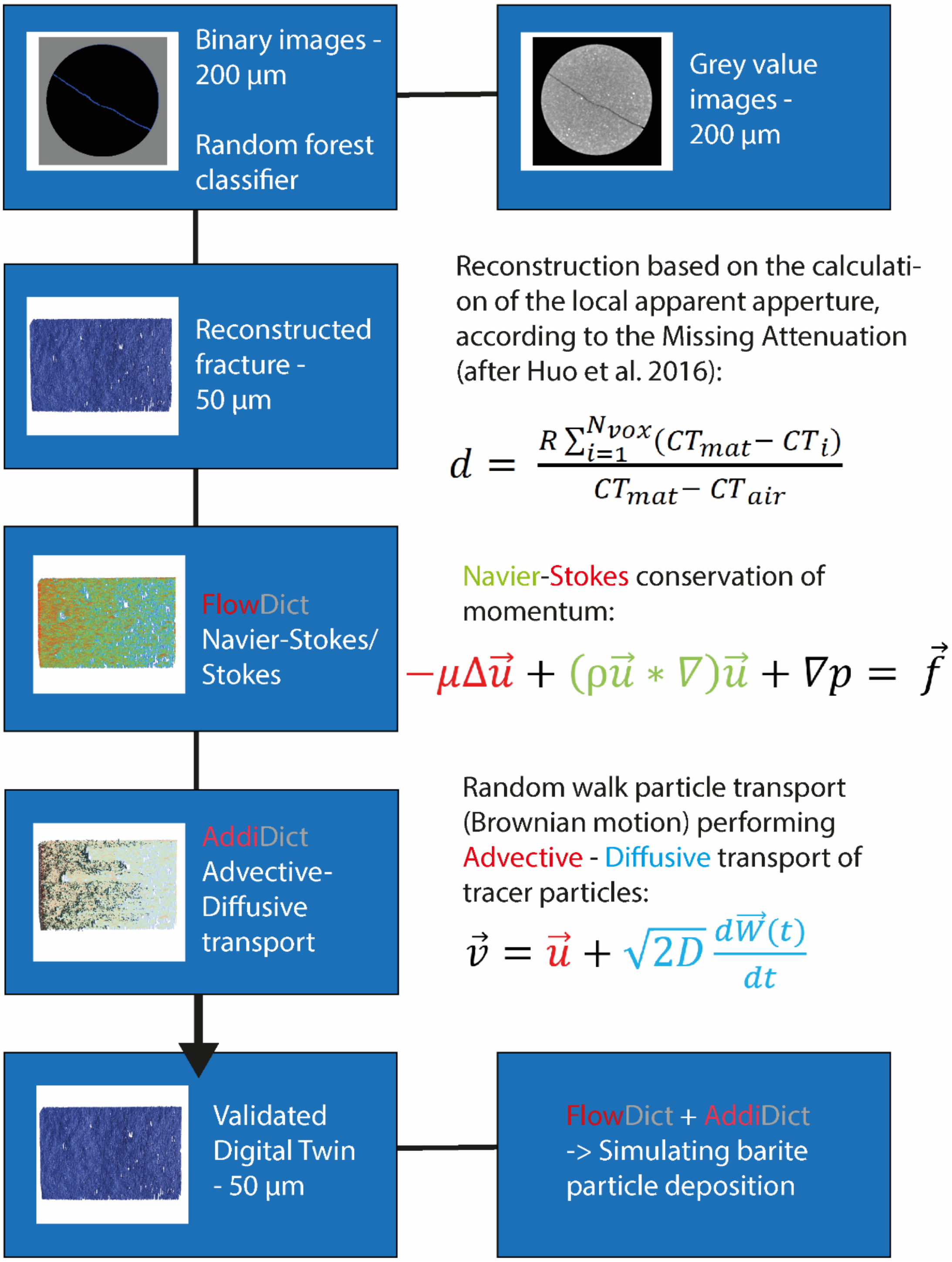

Binary images - 200 µm
Random forest classifier
Grey value images - 200 µm
Reconstructed fracture - 50 µm
Reconstruction based on the calculation of the local apparent appperture, according to the Missing Attenuation (after Huo et al. 2016):
$d = \frac{R\sum_{i=1}^{N_{vox}}(CT_{mat} - CT_i)}{CT_{mat} - CT_{air}}$
FlowDict Navier-Stokes/ Stokes
Navier-Stokes conservation of momentum:
$-\mu\Delta\vec{u} + (\rho\vec{u} * \nabla)\vec{u} + \nabla p = \vec{f}$
AddiDict Advective-Diffusive transport
Random walk particle transport (Brownian motion) performing Advective - Diffusive transport of tracer particles:
$\vec{v} = \vec{u} + \sqrt{2D}\frac{d\vec{W}(t)}{dt}$
Validated Digital Twin - 50 µm
FlowDict + AddiDict -> Simulating barite particle deposition

Figure 21. Workflow to reconstruct the validated digitized fractures
The segmented and the grey value XCT-images are obtained in the first steps. These images are used to calculate the local fracture apertures (by applying the CFMA method) and a resampled (in higher resolution) fracture skeleton. The skeleton is locally grown, according to the local aperture calculations, to obtain a reconstructed fracture. The reconstruction is validated by computing the flow field (and obtaining the fracture permeability) and tracer breakthrough curves. The validated DTs are used to model the hydro-mechanical barite deposition in the final section of this chapter.

4.2.2 Results and Discussion

The fractures Fle_8_S1 and Rem_1_S2 were reconstructed by applying the workflow described in sections 3.4 and subsection 4.2.1. For the Flechtinger sample, local CT_{mat} values and a growing limit of 80 µm were applied. The Remlinger sample was reconstructed by using local CT_{mat} values and a growing limit of 48 µm. The mechanical fracture apertures and relative roughness values were calculated with a modified version of the FTP algorithm (Cappuccio et al. 2020, also see subsection 3.5.3). Furthermore, permeabilities and breakthrough curves were calculated (see section 3.6). Fracture volumes were obtained by multiplying the number of voxels in the fracture with the voxel resolution cubed (i.e., $(50\ \mu m)^3$). The results were compared to experimental data (see section 3.5 for details regarding the experiments).

Table 2 (in subsection 3.5.3) presents the six correlations (equation 24 – 29) used to approximate the hydraulic aperture of the fractures. For the Fle_8_S1, a mean mechanical aperture of 175.5 µm (with a standard deviation of 90.1 µm) was calculated. The correlation formulas (Table 2) revealed hydraulic aperture values between 94 and 156 µm. The hydraulic aperture derived by the modified cubic law (Tsang 1992; Becker 2008) was 143.9 µm. For the Rem_1_S2, the mean mechanical aperture of 130.8 µm with a standard deviation of 38.3 µm was calculated. The hydraulic aperture was approximated to range between 89 and 126 µm. In comparison, the hydraulic aperture calculated from the modified cubic law gave a value of 93.7 µm. For both samples, the hydraulic apertures obtained from the modified cubic law and the correlations agree well. Furthermore, the mean mechanical apertures of the DTs are greater than the corresponding hydraulic apertures, which also agrees with literature data (also see section 2.1 and Figure 23).

For the Fle_8_S1 and Rem_1_S2, permeabilities of 1.18E-09 and 0.543E-09 m^2 were calculated (using the FlowDict module of GeoDict2020), respectively. As a comparison, the experimental data and the hydraulic aperture obtained by the modified cubic law (Tsang 1992; Becker 2008) were used to calculate the permeability (de Marsily 1986). The permeability of the Fle_8_S1 was 31.6 % smaller than the permeability based on the experimental data (1.73E-09 m^2). For the Rem_1_S2, a deviation of 25.7 % of the permeability based on experimental data (0.731E-09 m^2) was determined. In order to assess if this deviation is acceptable, the error of the permeability calculation (based on the experimental data) was considered. Due to simplified assumptions, permeabilities approximated with the cubic law are known to deviate from permeabilities from real fractures, depending on the individual geometry (Zimmerman and Bodvarsson 1996). Usually, the cubic law approximation overestimates the permeability (Xiong et al. 2011). E.g., Kluge et al. (2017) reported a permeability deviation of 50 – 80 % for their fractured samples. The modified cubic law approximation (equation 21) by Becker (2008) was tested on various fractured samples. This approach showed deviation usually smaller than 10 % (up to 21 %) for the hydraulic aperture. With this error range, the permeability calculated (equation 22) with this hydraulic aperture would be expected to usually deviate less than 21 % (up to 46 %) from the actual permeability. For the following study, this error range is assumed to be acceptable. The calculated permeabilities deviate 25.7 and 31.6 % from the permeabilities obtained from the modified cubic law approximation and, thus, are within this acceptable range. Furthermore, most simulated permeabilities are smaller than the cubic law approximation, which also agrees with the results reported in the literature (Xiong et al. 2011).

The calculated aperture values were used to estimate the fracture volume by multiplying the mean mechanical aperture with the approximated fracture length and width (i.e., the length and diameter of the core samples). The fracture volumes were also approximated by using the calculated hydraulic apertures. This illustrates the difference in estimated volumes when the mean mechanical aperture is unknown, and instead, the hydraulic apertures are used. As a comparison, the volume of the 3-D models was determined by multiplying the voxel amount with the resolution cubed (i.e., $(50\ \mu m)^3$). Since the length of the 3-D models was cropped during the processing (see section 4.1 and Table 6), the reduced length was used in the volume calculations. The volume of the Fle_8_S1 was approximated as $2.47 cm^3$ (with a mean mechanical aperture of 175.5 µm), which agrees well with the volume of the 3-D model ($2.55\ cm^3$). The volumes estimated with the hydraulic apertures range from $1.32 – 2.19\ cm^3$ (with an estimated volume of $2.03\ cm^3$, based on the hydraulic aperture obtained by the modified cubic law). Therefore, the volumes estimated with the hydraulic apertures deviate by about 14 - 48 % from the volume of the 3-D model. For the Rem_1_S2, a volume of $1.82\ cm^3$ was estimated from the mechanical aperture (130.8 µm), which was similar to the volume of the 3-D model ($1.86\ cm^3$). Volumes estimated from the hydraulic apertures deviated from these values by about 5 – 33 % (1.24 – $1.76\ cm^3$). With the hydraulic aperture derived from the laboratory experiments (93.66 µm), a fracture volume of $1.31\ cm^3$ was estimated.

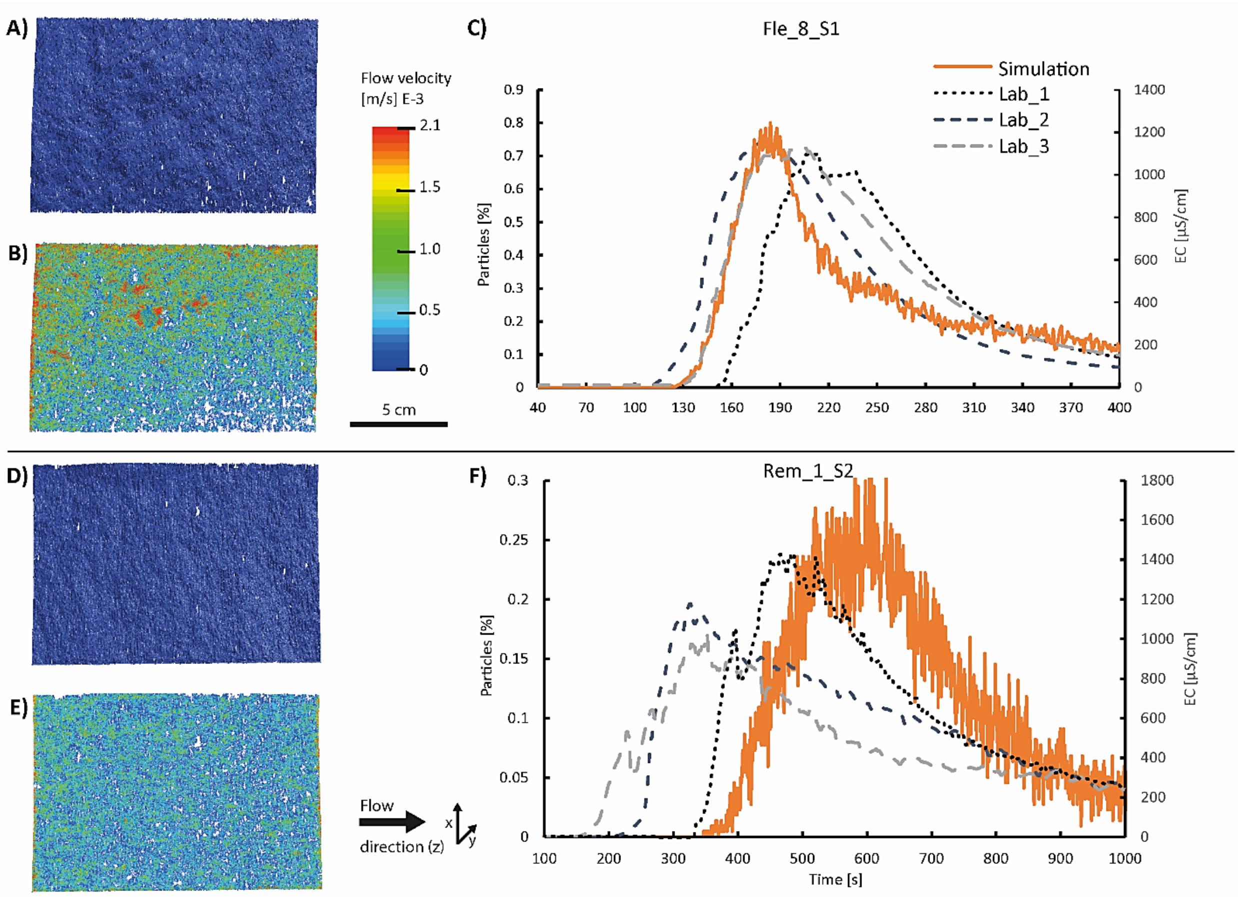


Figure 22. Result of the reconstruction of the samples Fle_8_S1 and Rem_1_S2
*The structures **(A & D)**, calculated flow fields **(B & E)**, and the simulated breakthrough curves (orange curves in **C, F**) are shown. As a comparison, for both samples, the breakthrough curves obtained by laboratory experiments are also presented.*

Figure 22 presents the reconstructed structures (A & D), the calculated flow fields (B & E), and the simulated breakthrough curves (C & F) of the Fle_8_S1 and Rem_1_S2 samples (similar comparisons for the other eight fractures can be found in the appendix, see Figure 50 - 53). The simulated breakthrough curves are not smooth since they are based on the variable number of particles that leave the fracture at each timestep. Increasing the number of particles, which represent the tracer, would smoothen the curves. However, the performance and simulation time depend on the size of the structure and the number of particles. Due to the large size of the reconstructed structures, fewer particles (~ $1x10^5$) could be simulated to maintain a reasonable simulation time.

In order to evaluate the breakthrough curves, three parameters (time of breakthrough, peak height, and $t_{50\%}$, the time until 50 % of the particles are detected) were compared with the values obtained from the experimental breakthrough curves. Arithmetic mean values and the 3σ confidence intervals were calculated (as $\bar{y} \pm \frac{3\sigma}{\sqrt{n}}$, with the number of observations (n = 3), standard deviation (σ), and mean value ($\bar{y}$) of each parameter) and compared with the simulated breakthrough curves. For the Fle_8_S1, simulated breakthrough time, peak height, and $t_{50\%}$ of 123 s, 0.79 %, and 241 s were obtained, respectively (Table 7). All three parameters lay in the 3σ confidence interval of the corresponding parameters of the experimental breakthrough curves, which were calculated as 97 – 165 s (breakthrough time), 0.61 – 0.79 % (peak height), and 199 – 269 s ($t_{50\%}$). For the Rem_1_S2 values of 359 s, 0.24 % and 668 s were determined for the breakthrough time, peak height, and $t_{50\%}$, respectively. The comparison with the experimental breakthrough curves revealed that the parameters of the simulated curve lay within the 3σ confidence interval of the breakthrough time (106 – 383 s), peak height (0.14 – 0.26 %), and $t_{50\%}$ (576 – 688 s). Therefore, the match between the simulated breakthrough curves and the experimental data was plausible, and the overall transport behaviour could be simulated with the reconstructed fractures.

Deviations between the simulations and the experiments could be caused by various factors, including:

(i) *The simplified particle injection function*. From the experiments, only the particle injection into the inflow tubes (not into the fractures) was determined. Since the simulated particles had to be created in the fracture, this injection function could not be used to reproduce the experimental conditions. Furthermore, due to the different positions of the fractures to the inflow tube, the particle injection function into the fractures is expected to differ for each sample. For these reasons, as a simplification, the particles were created simultaneously for each simulation. However, since simulations with varying particle injection functions (continuous release over 20 s, the Gaussian distribution release function (mean = 5 s, std = 1 s)) revealed only a minor deviation in the breakthrough curves, this effect might be neglectable.

(ii) *Fracture roughness*. Several studies (e.g., Zou et al. 2017; Stoll et al. 2019; Dou et al. 2019) demonstrated the importance of multiscale fracture roughness on the flow regime, e.g., enhancing flow heterogeneity and dispersion of transported particles. Due to the small fracture apertures, the roughness can have a significant influence on the flow, which results in deviations of simulated breakthrough curves from experimental data. For example, the breakthrough time was shown to deviate with higher roughness. The roughness may not be captured correctly (e.g., roughness below voxel resolution is not captured in the model) because of the limited spatial resolution and errors in the local aperture calculation.

(iii) *Resolution*. The resampled resolution (i.e., 50 µm) is still in a similar scale as the fracture apertures, which leads to three error sources. (i) During the reconstruction workflow, the fracture must be grown in 50 µm steps and therefore, aperture heterogeneity (and flow heterogeneity) deviated from the actual samples. (ii) Errors in the aperture calculation can have a considerable influence on the structure since aperture deviations can result in errors in local fracture growth, which are in the order of one voxel. (iii) Areas with a low apparent aperture might not be imaged in the 200 µm XCT-scans and, therefore, might not be included in the resampled DTs. Since the Remlinger fractures have a lower mean arithmetic aperture than the Flechtinger fractures, the resolution effects might have a more significant influence on the Remlinger DTs.

Although limitations of the presented workflow were concluded, the validation of both reconstructed exemplary samples demonstrated that proper DTs can be obtained by the proposed method. Compared to laboratory data, the reconstructed fractures showed similar properties in terms of fracture volume, hydraulic and mechanical aperture, permeability, and particle breakthrough behaviour. Therefore, each of the eight other samples was also reconstructed in the same manner, using the parameters shown in Table 6.

Table 6. Parameters applied during the reconstruction workflow
The parameters were applied to obtain the most suitable DT for each sample. The fitting was done by comparing simulated breakthrough curves with breakthrough curves from lab experiments. After the segmentation, each sample was cropped in Z-direction to remove ring artefacts. The final length of the models is provided in voxels (with a size of (50 µm)³). Voxels in the 50 µm skeleton with an aperture smaller than the applied growing limit did not grow in the skeleton growing phase. The subsequent limits are in 50 µm steps (i.e., for a grow limit of 75 µm, voxels with apertures between 75 - < 125 µm grow one voxel, 125 - < 175 µm grow two voxels, etc.).

Sample		**Length in Z-direction [voxel]**	**CT_{mat}**	**Growing limit**
Flechtinger	7_S1	2832	Local CT_{mat}	< 40 µm
	8_S1	2836	Local CT_{mat}	< 80 µm
	8_S2	2876	Global CT_{mat} = 3672	< 120 µm
	9_S1	2876	Global CT_{mat} = 3686	< 50 µm
	9_S2	2828	Local CT_{mat}	< 57 µm
Remlinger	1_S1	2896	Global CT_{mat} = 3585	< 70 µm
	1_S2	2804	Local CT_{mat}	< 48 µm
	1_S3	2804	Local CT_{mat}	< 35 µm
	2_S1	2864	Local CT_{mat}	< 30 µm
	2_S3	2776	Local CT_{mat}	< 55 µm

Figure 23 presents the aperture comparison for each reconstructed fracture. Hydraulic apertures were estimated with various conversion formulas (Table 2), using the mean mechanical aperture and the corresponding standard deviation. The hydraulic aperture was also estimated from experimental data by applying the modified cubic law approach (equation 21). The values of eight fractures agreed well with the range computed by the conversion formulas. For the other two fractured samples (Remlinger), the value from the modified cubic law deviated by about 18 % and 7.5 % from the lowest value of the calculated range. However, estimations from the modified cubic law were reported to deviate up to 21 % from measured hydraulic apertures, as elaborated above. Thus, the hydraulic apertures from these DTs are also plausible. Due to the high standard deviation (compared to the mean mechanical aperture), the correlation formula proposed by Xie et al. (2015) resulted in negative hydraulic apertures for two fractures (Fle_8_S1, Fle_8_S2). Therefore, these two values are not shown in Figure 23. Furthermore, the comparison of fracture apertures from the DTs showed that the Flechtinger fractures are wider than the Remlinger fractures. This observation agrees with the experimental data (Table 7). Computed fracture volumes also agreed well with the estimation from laboratory data (Table 7). Furthermore, for each reconstruction, the computed permeabilities lay within the acceptable error range (see above for the acceptable deviations of these parameters). Reynolds numbers were calculated according to equation 7, assuming the mean mechanical aperture as the characteristic length (Stoll et al. 2019). A fluid velocity of $1\text{x}10^{-3}$ m/s was used for the calculation, which corresponds approximately to the mean flow velocities calculated in the fractures and calculated in the laboratory experiments. Since all Reynolds numbers are smaller than 1, the viscous forces can be neglected, and the Stokes equation is suitable for the calculation of the flow fields. Overall, each reconstruction exhibits realistic hydraulic apertures, permeabilities, and fracture volumes.

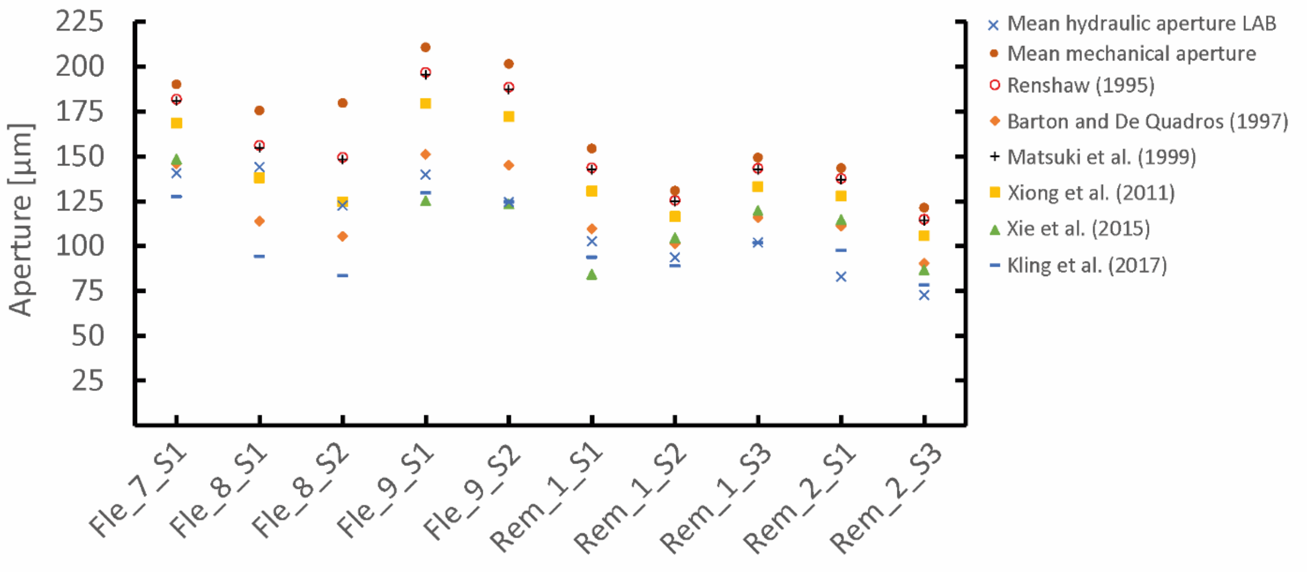


Figure 23. Estimation of the hydraulic aperture
Various conversion formulas from the literature were used to approximate the hydraulic aperture. The mean mechanical aperture and the calculated hydraulic aperture (from the laboratory experiments) are also shown for each sample. Due to the high standard deviation compared with the mean aperture, the approximation of Xie et al. (2015) resulted in negative hydraulic apertures for two fractures (Fle_8_S1, Fle_8_S2). These values are not shown here.

Table 7. Overview of the properties of all reconstructed DTs
Comparison of estimated fracture volumes (based on the hydraulic aperture) and permeabilities, based on lab data, with the volumes and permeabilities of the DTs. For the volume calculations, the length of the cropped DT was used. Additionally, the roughness and standard deviation of the mean mechanical aperture are presented.

Parameter	**Flechtinger**					**Remlinger**				
	7_S1	**8_S1**	**8_S2**	**9_S1**	**9_S2**	**1_S1**	**1_S2**	**1_S3**	**2_S1**	**2_S3**
Fracture volume DT [cm³]	2.78	2.55	2.76	3.23	2.89	1.95	1.86	2.16	2.10	1.74
Estimated fracture volume [cm³]	1.98	2.03	1.76	2.00	1.75	1.48	1.31	1.42	1.18	1.00
Deviation from estimate [%]	40.1	25.6	56.7	61.7	65.1	31.9	42.3	52.3	77.7	74.2
Roughness DT [-]	0.305	0.513	0.666	0.382	0.377	0.394	0.293	0.291	0.293	0.339
Standard deviation of aperture [µm]	57.9	90.1	119.6	80.5	75.9	60.9	38.3	43.5	42.1	41.2
Permeability DT E-09 [m²]	1.44	1.18	0.83	1.57	1.40	0.573	0.543	0.649	0.568	0.342
Permeability LAB E-09 [m²]	1.52	1.73	1.25	1.54	1.25	0.879	0.731	0.867	0.575	0.441

Deviation from LAB [%]	5.2	31.6	33.5	2.1	11.8	34.7	25.7	25.1	1.2	22.4
Reynolds number [-]	0.19	0.18	0.18	0.21	0.20	0.15	0.13	0.15	0.14	0.12

Breakthrough curves were simulated with each reconstructed fracture. Table 8 presents three parameters (breakthrough time, peak height, and $t_{50\%}$) of the simulated and measured breakthrough curves. Since at least three tracer experiments were conducted for each sample, Table 8 shows the mean values and the 3σ confidence interval for each parameter. Overall, the simulated curves of the Flechtinger fractures showed a similar behaviour as the measured curves, with deviations that lay within the computed 3σ confidence interval. However, 2 - 3 parameters from the breakthrough curve of four Remlinger fractures exhibited larger deviations from the experimental data. The deviations could be explained by the resolution and the small apertures of these fractures due to the following reasons: (i) Calculated apertures could only be reconstructed in 50 µm steps, resulting in a considerable influence of the growing limit and errors in the aperture calculation. The 50 µm spatial resolution might be too coarse to accurately resolve the variations of the real fracture apertures, which also influences the computed flow fields (ii) Small scale roughness cannot be represented with a resolution of 50 µm (iii) Small apertures (< 50 µm) might not be segmented in the original medical XCT images (due to strong blurring in the 200 µm resolution), leading to more contact areas in the reconstructed fractures (in 50 µm resolution), compared to the real sample. Two parameters (growing limit and local or global CT_{mat} value) were varied during the reconstruction workflow to counteract these error sources. The breakthrough curve is the most sensitive characteristic used as validation in this study and is, therefore, the first property that is expected to show deviations when the model starts to deviate from the real samples. Overall, the large-scale parameters (permeability, fracture volume, mean aperture) were captured well with each reconstructed DT. Deviations in the breakthrough curve indicate that details in the fracture structure (e.g., roughness, the variety of apertures) were not captured well for all samples, resulting in differences in the computed flow fields (and tracer transport). In that sense, it was possible to reconstruct six fractures with good quality, but four Remlinger samples showed larger deviations. Since the Remlinger fractures exhibit lower mean apertures than the Flechtinger fractures, the influence of the errors mentioned above could be greater and thus explain why most DTs of the Remlinger fractures showed large deviations from the experimental data.

To summarize, with the presented workflow, it was possible to reconstruct each fracture at a higher resolution. While large-scale parameters (permeability, volume, mean aperture) fitted well with experimental data, more sensitive small-scale parameters (from breakthrough curves) showed deviations for some DTs. Six fractures showed a good match with all available experimental data. Four reconstructed (Remlinger) fractures showed larger deviations of two or three parameters of the simulated breakthrough curves and therefore revealed a lower quality match. Due to the small apertures, these deviations could be mainly caused by insufficient representation of roughness and the variations in the aperture field (due to the low spatial resolution). Reconstructing the fractures at different resolutions was not feasible in this study because particle migration simulations with the applied resolution of 50 µm were already at the computational limit. Furthermore, the reconstruction at lower resolutions (i.e., 100 µm or 200 µm) would not lead to meaningful structures since most of the fracture area would not be well represented due to the high fraction of apertures with a low aperture (in these resolutions, represented by only one or two voxels).

Table 8. Comparison of parameters describing simulated and real breakthrough curves
Parameters of the simulated breakthrough curves are highlighted in red if they are not included in the 3 σ confidence interval of the experimental curves.

Parameter	**Flechtinger**					**Remlinger**				
	7_S1	**8_S1**	**8_S2**	**9_S1**	**9_S2**	**1_S1**	**1_S2**	**1_S3**	**2_S1**	**2_S3**
Breakthrough simulated [s]	116	123	122	95	133	176	359	295	372	587
Breakthrough LAB mean [s]	83	131	119	100	142	129	244	171	253	264
3σ interval [s]	[43 – 122]	[97 – 165]	[90 – 149]	[51 – 149]	[128 – 156]	[80 – 178]	[106 – 383]	[117 – 225]	[146 – 359]	[251 – 277]
Peak height simulated [%]	0.93	0.79	0.47	0.48	0.71	0.6	0.24	0.4	0.24	0.22
Peak height LAB mean [%]	0.89	0.70	0.36	0.49	0.83	0.27	0.20	0.23	0.20	0.05
3σ interval [%]	[0.77 – 1.01]	[0.61 – 0.79]	[0.22 – 0.50]	[0.45 – 0.54]	[0.68 – 0.98]	[0.24 – 0.31]	[0.14 – 0.26]	[0.15 – 0.31]	[0.15 – 0.24]	[0.05 – 0.06]
$t_{50\%}$ [s]	199	241	284	227	228	301	668	491	737	960
$t_{50\%}$ LAB mean [s]	164	234	325	248	233	387	632	529	601	1558
3σ interval [s]	[118 – 210]	[199 – 269]	[276 – 373]	[191 – 304]	[224 – 242]	[369 – 405]	[576 – 688]	[343 – 716]	[524 – 678]	[1409–1706]

4.2.3 Comparison to segmented fractures

The reconstruction workflow was applied, and the obtained fracture models were validated with experimental data. To further assess the applied workflow, this subsection compares the reconstructions (in 50 µm resolution) with the RF segmented fractures (in 200 µm resolution) of the Fle_8_S1 and Rem_1_S2. As a reference, the experimental data is also included for comparison. Similar differences described for these two samples were also observed for the eight other samples (see Figure 54 and 55 in the appendix).

Table 9. Properties of the reconstructions and segmentations of the reference samples
Comparison of the permeability, mean aperture, volume, and roughness of the reconstructed (50 µm resolution) and segmented fractures (200 µm resolution) Fle_8_S1 and Rem_1_S2.

Parameter	Fle_8_S1 Reconstruction (50 µm)	Fle_8_S1 Segmentation (200 µm)	Rem_1_S2 Reconstruction (50 µm)	Rem_1_S2 Segmentation (200 µm)
Permeability E-09 [m²]	1.18	46.8	0.543	21.9
Permeability LAB E-09 [m²]	1.73		0.731	
Deviation from LAB [%]	31.6	2604	25.7	2897
Mean mechanical aperture [µm]	175.5	827.2	130.8	571.6
Hydraulic aperture [µm]	143.9		93.66	
Deviation from LAB [%]	21.9	474	39.7	510
Volume [cm³]	2.55	11.88	1.86	8.00
Estimated volume [cm³]	2.03		1.31	
Deviation from LAB [%]	25.6	485	41.9	441
Roughness [-]	0.513	0.216	0.293	0.221

Table 9 presents the parameter comparison of the segmented and reconstructed fractures. The simulated permeabilities of the segmented fractures were about one order of magnitude higher than the permeabilities of the corresponding reconstructions. Consequently, major deviations from the laboratory data were observed. The segmentations deviated over 2500 % from the experimental data, whereas the reconstructions deviated by less than 32 %, which was still in the acceptable range of error (see subsection 4.2.2). Similarly, the mean mechanical apertures and the fracture volumes were about five times larger than the values calculated from experimental data, with resulting deviations from 440 – 510 %. Again, the reconstructions showed many similar values with deviations within the plausible range (see subsection 4.2.2). Figure 24 illustrates the aperture distributions of both samples and in both resolutions. In order to allow for a better comparison, the calculated hydraulic apertures (from experimental data) are highlighted with a red line. The direct comparison in Figure 24 highlights the pronounced difference in the distributions. Most of the measured apertures in the reconstructed fractures were below 200 µm (with mean values close to the hydraulic aperture), whereas only a few apertures of the segmentations lay in this range. Furthermore, the aperture distribution of the reconstruction exhibited a typical shape for sandstone fracture aperture distributions (e.g., Chen et al. 2020; Kluge et al. 2021). Overall, the reconstruction workflow provided considerably more realistic DT models of the real fractures.

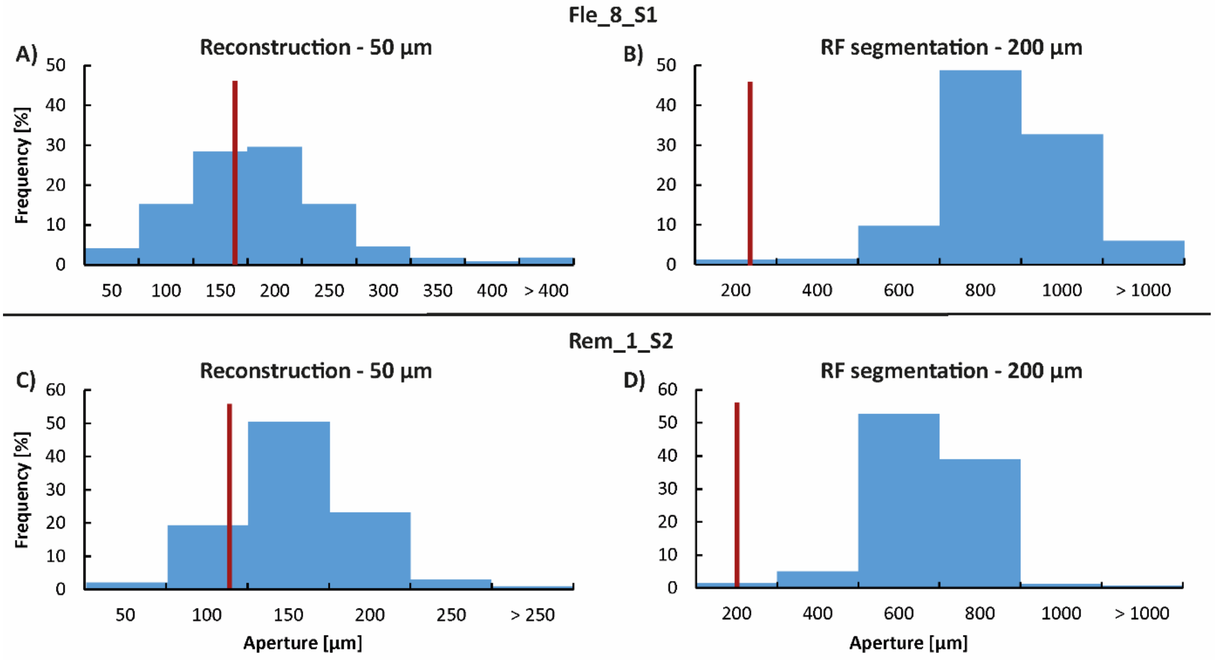


Figure 24. Apertures of the reconstructions and segmentations of the reference samples
*Comparison of the mechanical aperture distribution between the reconstructed fractures (with 50 µm spatial resolution, **A** and **C**) and the segmented fractures (with 200 µm spatial resolution, **B** and **D**) using the RF classifier. The red line indicates the hydraulic aperture, calculated from experimental data. For the histogram classes, the bin size was chosen to be equal to the corresponding spatial voxel resolution.*

4.2.4 Discussion and conclusion

The advantages of the RF segmentation were demonstrated in section 4.1. However, deviations from experimental breakthrough curves were observed and attributed to the PVE. The segmented fractures were used as the foundation for a novel reconstruction workflow to obtain fracture models in higher resolution and analyse the influence of the PVE. The reconstructed fractures were validated with experimental data (mean aperture, permeability, fracture volume), and observed deviations were within a plausible range of error. The comparison of breakthrough curves demonstrated that six reconstructed fractures represented plausible DTs of the real samples. However, four reconstructions showed significant deviations from the experimental breakthrough curves. These deviations may be mainly attributed to the limited resolution and the insufficient representation of fracture roughness in the models. A larger influence of these errors could be expected in the smaller Remlinger fractures and, thus, could explain why only these DTs showed larger deviations. The applied resolution of 50 µm was already at the computational limit; hence, it was not feasible to further enhance the resolution to analyse the possible improvement in the corresponding fracture reconstruction. To fully assess the reliability of the presented workflow, it should be considered to analyse the impact of resolution and the uncertainty of the aperture calculation (by the CFMA) in future studies. Studies reported deviations of about 20 – 60 µm of the CFMA aperture calculation, depending on the sample, which indicates that further resampling might be limited by this calculation error (Huo et al. 2016; Wenning et al. 2019).

For the segmentations, major deviations from experimental data were observed for permeabilities, apertures, and fracture volumes. A comparison of the reconstructed (in 50 µm resolution) with the segmented fractures (in 200 µm resolution) revealed the considerable improvement in the fracture models. All reconstructed fractures showed similar characteristics as observed in the conducted experiments. With the presented workflow, it was possible to considerably improve the digital fracture models obtained by the segmentation. This was possible without the need for additional data (which was not available in this study), which is required for alternative reconstruction methods reported in the literature (e.g., Ramandi et al. 2017; see section 2.3). For six fractures, a plausible match for all parameters (permeability, aperture, breakthrough curves) was achieved, whereas the breakthrough curve of the other four fractures significantly deviated from experimental data. Additionally, the analysis highlighted the importance of proper validation of the DT models. Previous studies only applied a limited number of parameters (like the permeability in Ramandi et al. 2017) to verify the presented fracture models. Results from this thesis indicate that additional parameters are required for verifying DTs fractures. Each reconstructed DT showed realistic large-scale parameters (permeability, apertures); however, only the comparison of the small-scale parameters (from the breakthrough curve) revealed larger deviations from the experimental data for most Remlinger fractures. Since the flow field in a fracture highly depends on the complex geometry (e.g., roughness, aperture heterogeneity, contact zones; He et al. 2021), a sufficient spatial resolution is needed to represent these complexities in a DT. In the presented models, the (resampled) spatial resolution was in a similar range as the fracture apertures, i.e., most apertures were represented by few voxels (< 5). Hence, the DTs did not exhibit the actual surface roughness (below voxel size) and the real aperture heterogeneity, leading to deviations in the breakthrough curves. For example, Saxena et al. (2018) concluded that at least ten voxels are needed to sufficiently resolve a pore throat diameter. Since deviations in surface geometry should have a bigger influence on the flow field in the smaller Remlinger fractures, this could explain the observed deviations in the breakthrough curves of four Remlinger samples.

Future studies might help to improve the quality of the reconstruction and quantify the limits of the workflow. Possible investigations could include: (i) The analysis of 3-D printed fractures with defined properties (e.g., a fracture with a constant aperture or a fracture with a defined aperture distribution with a large mean aperture, which allows to reconstruct meaningful DT in different resolutions). These samples could be imaged (e.g., by XCT) in different resolutions to compare the reconstructed DT to the segmentation in similar resolutions and the ground truth. (ii) The direct comparison of the result from the presented workflow with results of other reconstruction methods (e.g., the approach of Ramandi et al. (2017) or the method of Cappuccio et al. (2020) could be applied to reconstruct single fractures) or images obtained by other imaging techniques (e.g., SEM). This comparison could be conducted in 2-D (pixel by pixel) or 3-D. Such a comparison could also help to further quantify the error of the CFMA calculation. (iii) The error of the CFMA calculation should be quantified for different samples and resolutions to investigate the constraints of the method (e.g., by direct comparison with SEM images). (iv) The analysis of natural fractures with a larger mean aperture or a smaller length than the samples analysed in this thesis. Such fractures could allow to obtain meaningful reconstructed DT on various resolutions and, therefore, to analyse if the quality of the DT improves when the resolution during the reconstruction is enhanced by various factors. Additionally, such an analysis could provide insights on an optimal ratio of fracture aperture to the (enhanced) resolution. (v) A well-resolved fracture could be reconstructed in higher resolutions to investigate the influence of small-scale details (if the error in the CFMA aperture calculation is proven to be small enough to reliably resolve these details) in the fracture structure (roughness, small aperture zones) e.g., on the flow field. The fracture roughness might pose a valuable validation parameter.

4.3 Barite particle deposition modelling

This section investigates the migration of barite particles through the previously validated DT models of real fractured sandstones. A sensitivity analysis is conducted to examine the influence of various parameters on the number of deposited particles. The goal of this section is to analyse which settings (e.g., small or large flow rates, higher or lower temperatures) are more favourable for establishing a sustainable use of fractured geothermal reservoirs. The simulation parameters are based on laboratory experiments, which were conducted by the project partners at the RUB. Two successful flow-through experiments (using the constructed HPTC) with the injection of prepared barite particles were conducted. Details regarding the experimental procedure on the Fle_9_S1 can be found in Frank et al.

(2021). In the first subsection, suitable simulation parameters are determined, and simulation results are compared with results from the experiments. The second subsection presents a sensitivity study, where the influence of 6 parameters (flow rate, temperature, particle size, particle amount, adhesion forces, random seed) on the amount of particle deposition in two fractured samples (Fle_8_S1 and Rem_1_S2) is investigated. The sensitivity analysis is complemented by a comparison of particle and tracer transport and an analysis of the relationship between fluid flow velocity and particle deposition. This section is concluded by summarizing the results and formulating suggestions for minimizing particle deposition.

4.3.1 Base model setup and validation

To mirror the experimental conditions in the HPTC, the validated DT fractures obtained in the previous section and the simulation setups (see section 3.6 and subsection 4.2.1) were modified in the following way. The structures were flipped (i.e., the inflow was positioned at the top of the domain and the outflow at the bottom) to simulate a flow from top to bottom. For the flow field simulation, the temperature was set to 58°C, and a constant pressure drop was applied to enable a flow through the fractures. Similar to the tracer simulations, the Rem_1_S2 sample was used to calculate the pressure drop to the corresponding flow rate. Since each fracture has a different aperture distribution, different flow rates are produced at the same pressure drop (see Table 1). Thus, the calculated pressure drop for the Rem_1_S2 sample was reduced to achieve the same flow rate in the Flechtinger samples (e.g., a flow rate of 2.03225 mL/s resulted in the pressure drop of -520.0 Pa in the Rem_1_S2. The same flow rate in the Fle_8_S1 would be achieved with a pressure drop of -144.4 Pa since it was observed in the experiments that the measured flow rates differed by a factor of 3.6 (0.32 mL/s for the Rem_1_S2 and 1.14 mL/s for the Fle_8_S1) at the same pressure drop). A negative pressure drop was applied to achieve a flow from top to bottom.

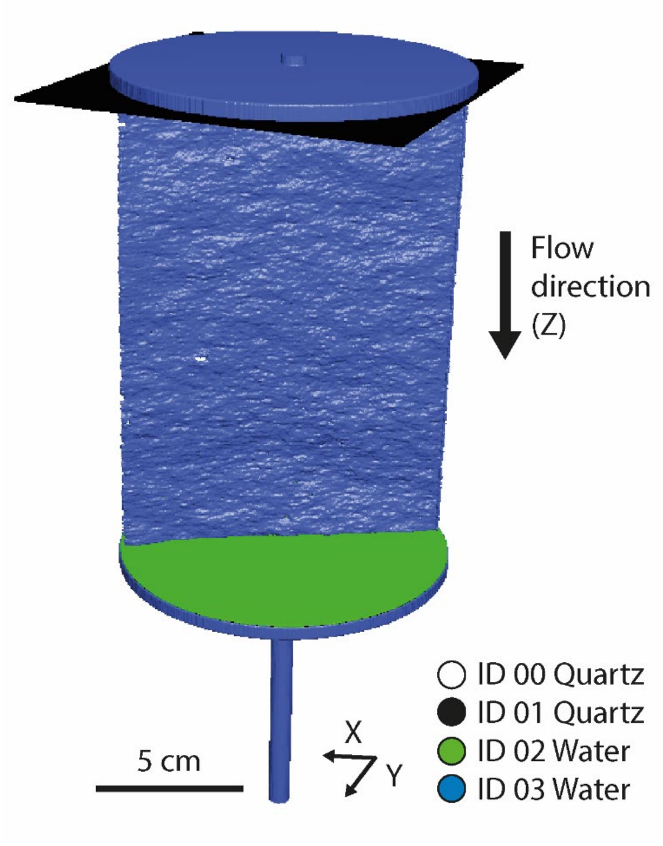


Figure 25. Setup for conducting the barite particle migration simulations
The DTs obtained in the previous step (section 4.2) were flipped, and the water flow was applied from top to bottom. The fracture, pipes, and in-/outflow areas are filled with water (ID 03, blue), and the matrix and background are composed of quartz (ID 00, invisible). Particles are created in the first ten slides of the fracture. This spawn area is encased by a different material (ID 01, quartz, shown in black) to avoid instant deposition of particles that are created close to the fracture walls. Particles are included in the breakthrough curves when they reach the outflow area below the fracture (ID 02, green). In this setup, particles are either transported through the fracture or deposited in the fracture.

Simulations with particles placed in the inflow tubing revealed that most barite particles were deposited at the rock interface instead of entering the fracture. Since this study aims to investigate the particle migration through the fracture and only a limited number of particles could be simulated, the number of particles entering the fractures had to be maximised. Therefore, the barite particles were directly placed into the fracture (within the first ten slides, with the starting velocity equal to the flow velocity in the corresponding voxel). The interaction of the particles with the surrounding rock matrix of this spawn area was simulated with the sieving model (the restitution value was set to 1) to avoid the instant deposition of the created particles. The particle interaction with the rock matrix enclosing the fracture was simulated with the Hamaker model, where the restitution was set to 0.9 and the Hamaker constant to 10^{-23} J. These values were chosen to achieve similar deposition patterns as observed in the experiments. Similar values were also applied in other modelling studies (e.g., Sadeghnejad et al. 2022). Particles that were transported through the fracture were registered directly at the outflow of the fracture (and considered in the breakthrough curves). 3x10^6 particles with a density of 4480 kg/m³ and sizes of 4 (40 % of the particles), 7 (30 %), 10 (20 %), and 14 (10 %) µm were simulated. The size distribution was chosen to approximate the measured size distribution of deposited particles in the Fle_9_S2 fracture (Figure 12). The simulation time was set to 500 s, with a time step of 1 s, periodic BC in tangential directions, open BC at the lower end, and reflective BC at the upper end of the domain. The particle motion was calculated by considering advection and diffusion (equation 12), where the diffusion coefficient of the particles was calculated with equations 13 and 14. Additional forces (e.g., electrostatic forces) were neglected to save computational resources. Breakthrough curves, mean collisions, and the number of escaped/deposited particles were tracked for each simulation.

After the laboratory experiments were conducted, the cores were dismounted, opened, and images of both fracture walls were taken. One side of each fractured core was cut into eight smaller pieces and analysed with a SEM by the project partners at the TUD and RUB universities. For the Fle_9_S2 sample, the size distribution of deposited barite particles was created (Figure 12), and for each sample (Rem_1_S2, Fle_9_S1, and Fle_9_S2), snapshots of the deposited particles were taken (examples shown in Figure 13). A quantitative analysis (exact locations, amount) of the deposited particles was not conducted by the project partners. Furthermore, the deposited particles were too small to be detected in CT-images (spatial resolution of 200 µm, with particle sizes up to 30 µm), and the particles did not illuminate when exposed to different light sources. Thus, only the photographs could be used for the comparison with the simulation results. The deposited barite particles could not be segmented in the images due to the following reasons: low quality and resolution of the images, uneven illumination, similar colour of barites and other minerals, and distinguishing injected barites and barites from the rock matrix was not possible. Therefore, only a qualitative comparison of deposition patterns was conducted.

For the qualitative comparison with the simulation results, the Fle_9_S1 and Rem_1_S2 samples were used. The flow-through experiments with these two fractures were conducted with approximately constant flow rates (in the saturation phase, i.e., before and during injection) and an injection of a prepared barite particle mix. During the experiment with the Fle_9_S2, the core was damaged, and the pump malfunctioned, causing highly variable flow rates. These conditions could not be simulated, and it was not clear when the core was damaged and how this affected the particle transport.

Therefore, only the experiments with the Fle_9_S1 and Rem_1_S2 could be simulated with the DT models.

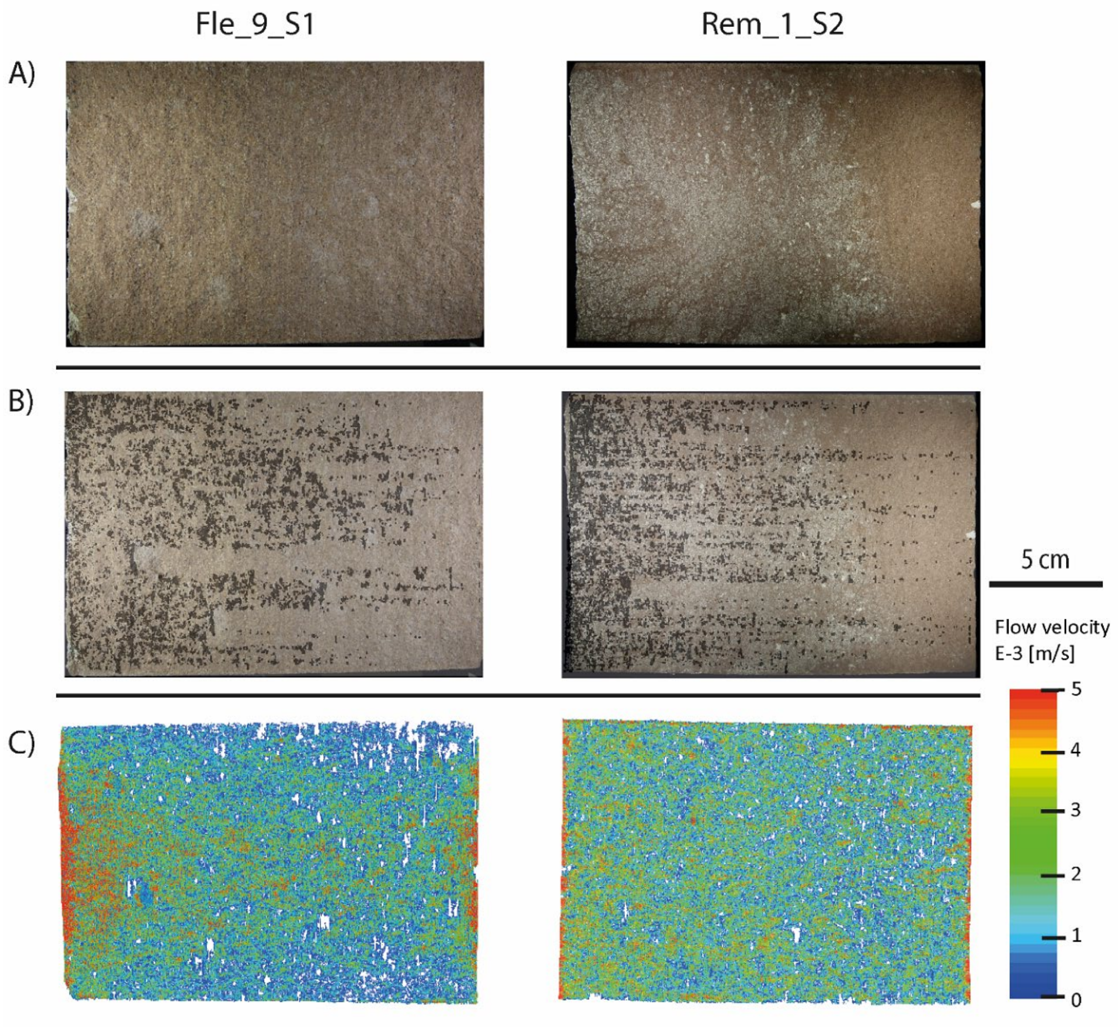


Figure 26. Qualitative comparison of the barite deposition results
***(A)** Photographs of one fracture half after the laboratory experiments were conducted. **(B)** The photographs are overlain with the deposited particles (in black, the size was increased by a factor of 50 for the visualisation) obtained from the simulations with the DTs. **(C)** Calculated flow fields with a flow rate of 2 mL/s (Fle_9_S1) and 1.32 mL/s (Rem_1_S2).*

Figure 26 presents the results for the Fle_9_S1 and Rem_1_S2 samples. The photographs of the actual cores (after conducting the experiments) are shown (Figure 26A) and overlain with the deposited particles obtained by the simulations (Figure 26B). For the Fle_9_S1 particle simulation, deposition across the whole fracture was observed. Zones with low deposition coincide with upstream contact areas between the fracture halves (Figure 26C). High deposition occurred at these contact zones or when the flow velocity was low. Similar patterns could be observed in the photograph, where deposition is indicated by an accumulation of white minerals. However, as elaborated above, the white spots are a mixture of deposited barite particles and minerals of similar colour that belong to the sandstone matrix. Therefore, the interpretation of the photographs is not unique and is of a qualitative nature. In the Fle_9_S1 core, accumulations of white minerals were observed across the core, and larger accumulations were in similar areas as particle depositions in the simulation (e.g., inflow area and contact zones between the fracture sides). For the Rem_1_S2, similar deposition patterns were observed in the simulation: high deposition at the inflow and at contact zones and low deposition behind contact zones. Furthermore, as observed in the photograph, most particles were deposited in the anterior area of the fracture.

As elaborated above, a quantitative comparison of laboratory results and the simulations could not be conducted. Qualitatively, both DTs showed similar deposition patterns as observed in the experiments. Thus, the applied simulation settings were used to perform the following sensitivity analysis, aiming to quantify the influence of various parameters on barite particle deposition. Future studies could evaluate similar experiments in a quantitative way, by a complete SEM imaging of the fracture halves, where barites can be identified by energy-dispersive X-ray spectroscopy. Other approaches could include the staining of the prepared barite particles with a distinct colour or choosing a smaller fractured core, which would allow for a higher resolution of the XCT-scans. Due to the high density of the barite particles, it might be possible to detect depositions at higher spatial resolutions.

4.3.2 Sensitivity analysis

As qualitatively shown in the previous subsection, the model setups with the DTs resulted in similar particle deposition patterns as in the experiments. Hence, these parameters were used to simulate the base model for the sensitivity analysis. In order to isolate the influence of particle size, a single diameter (10 µm in the base model) was used in the simulations. The same samples, which were used in the previous sections for demonstrating the segmentation and reconstruction workflow, were selected to conduct the sensitivity analysis: one Flechtinger (Fle_8_S1) and one Remlinger sample (Rem_1_S2). As stated above, the flow field simulations on the Rem_1_S2 were conducted to calculate the corresponding pressure drop, which is required to have a specific flow rate at a certain temperature. For the flow field simulations of the Fle_8_S1, these pressure drops were divided by a factor of 3.6 to achieve the same flow rates (all flow field simulations were conducted by applying a pressure drop).

The values for the parameters of the base model and their corresponding range for the sensitivity analysis are summarized in Table 10. Six parameters (flow rate, temperature, particle size, particle amount, adhesion forces, random seed) were varied to investigate their influence on the hydro-mechanical barite particle deposition in the fractures. The sensitivity study was conducted by using the base model as a reference. The effect of each parameter was isolated by keeping five parameters constant while varying the value of the sixth one. The flow fields for both base models are presented in Figure 27.

Table 10. Summary of base model parameters and the ranges for the sensitivity analysis

Parameter	Base simulation	Range for sensitivity analysis
Flow rate [mL/s]	2	0.5 - 8
Temperature [°C]	58	38 - 78
Particle amount [-]	$3x10^6$	$0.5 – 6x10^6$
Particle size [µm]	10	1 - 42
Hamaker constant [J]	10^{-23}	$10^{-24} – 5x10^{-22}$

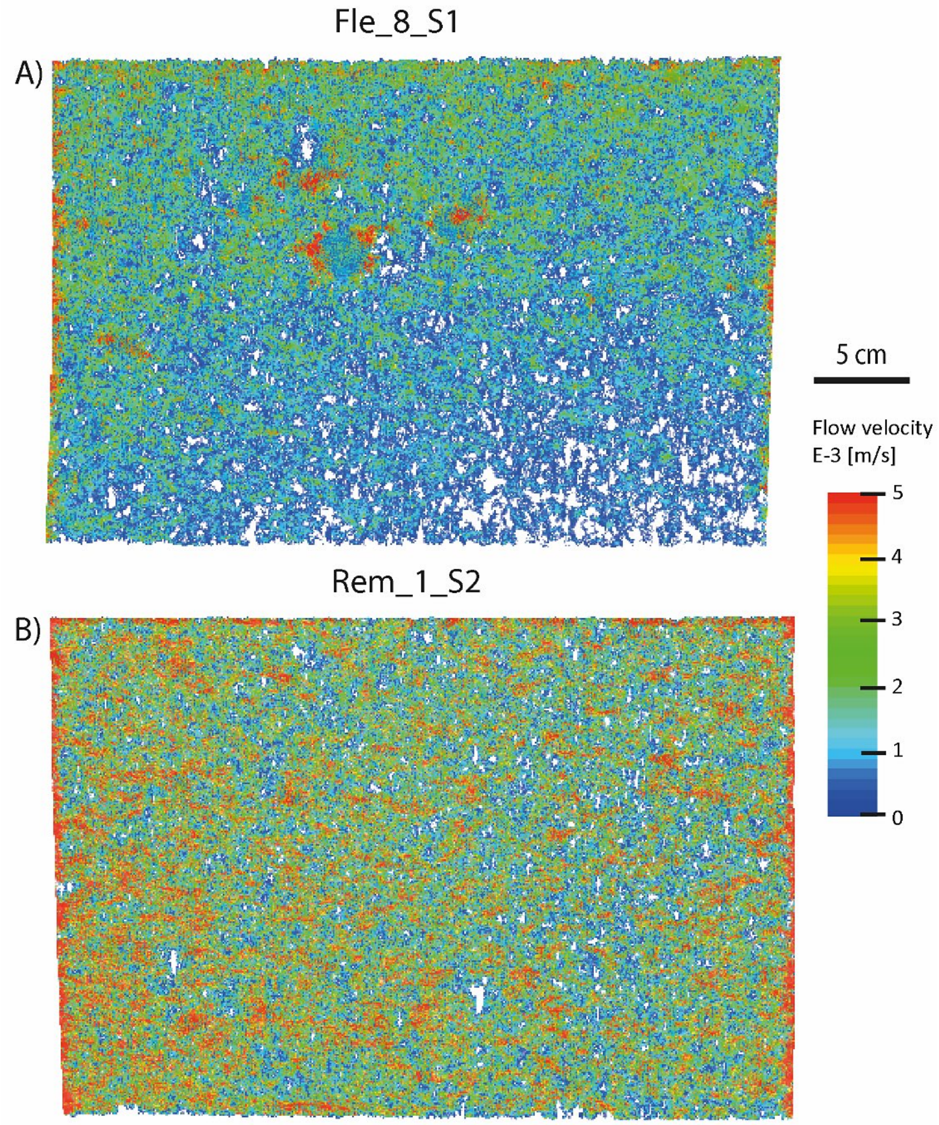


Figure 27. Computed flow fields for the base models
The results for the Fle_8_S1 are shown in ***(A),*** *and the results for the Rem_1_S2 are shown in* ***(B)****.*

The sensitivity analysis is organized as follows: the first subsection is dedicated to quantifying the numerical error of the simulations. Three possible error sources were considered: the random seed for particle placing, the number of simulated particles, and the error bound of the flow field calculation. The subsequent subsections present the results for the sensitivity analysis on each parameter and a discussion of their corresponding influence. The second-to-last subsection focuses on the comparison of barite particle migration with the migration of tracer molecules. In the last subsection, the fluid flow velocities in the voxels where particles were deposited are analysed. Most subsections show the results from both fractures, which are presented in the form of cumulative breakthrough curves and the number of particles that were caught in the fracture.

Numerical error

In the first step, the numerical error of the simulations was assessed on the Flechtinger sample. The random seed was varied between 75 and 84 to investigate the influence of different particle start positions. The number of particles caught in the fracture and their fraction, compared to the 3x10^6 starting particles, were compared. The particle amount varied between 0.5 – 6x10^6 and the fraction of caught particles was compared (Figure 28).

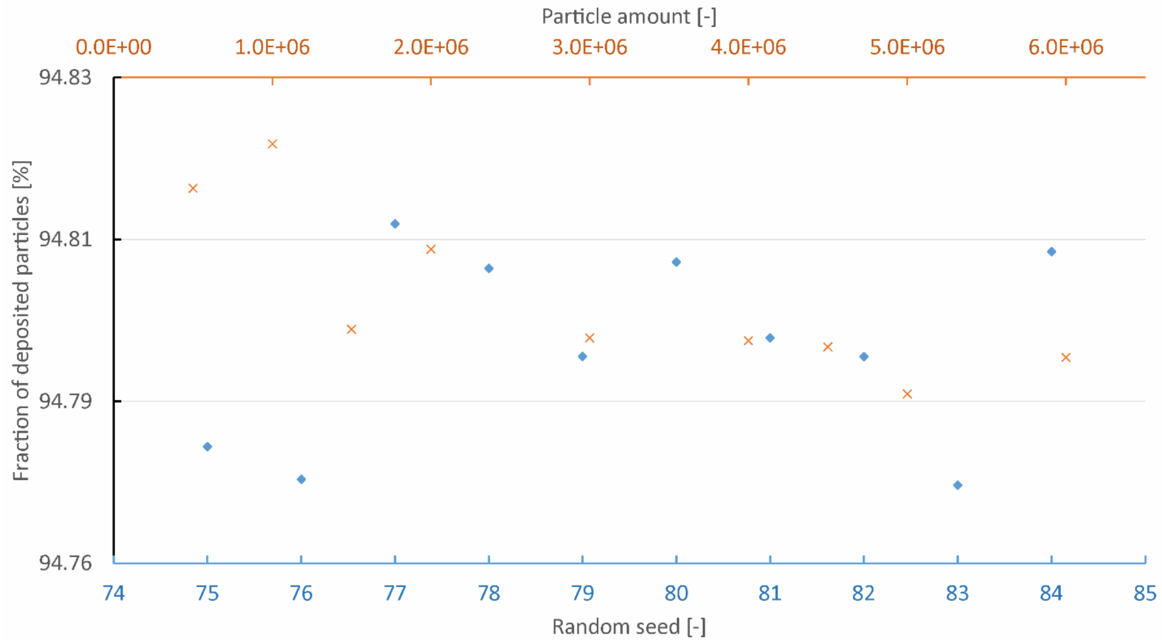


Figure 28. Assessment of the numerical error of the barite migration simulations
The random seed and the particle amount were varied in the simulations with the Fle_8_S1.

For the random seed sensitivity analysis, the fraction of deposited barite particles varied between 94.774 and 94.813 %, with a mean value of 94.795 % and a standard deviation of 0.014 % (3σ = 0.014 %, with n = 10). On average, 156 153 particles were detected at the outflow of the fracture. The particle amount simulations resulted in deposited particle fractions of 94.788 – 94.825 %, with a mean of 94.802 % and a standard deviation of 0.012 % (3σ = 0.012 %, with n = 9). The sum of both 3σ ranges was 0.026 %, which corresponded to a deviation of 156153 ±787 particles escaping the fracture. The error bound of the flow field calculation was set to 5 %, which was an order of magnitude higher than the deviations because of different particle starting positions (random seed) and particle amount. A deviation of 5 % (which corresponded to 156 153 ±7808 particles) was detected at the outflow. Hence, the influence of different random seeds and the particle amount can be neglected, and a numerical error of 5 % was assumed for calculating the error bars in the following results.

For the following simulations, the particle amount was set to $3x10^6$ since the fraction of deposited particles only slightly changed with higher numbers, and enough particles were simulated to reliably determine the number of deposited particles. For both samples, the arithmetic mean value of the number of particles escaping the fracture was determined, and the random seed was selected, which resulted in the closest value to this mean. The following constant random seeds were applied for the simulations presented below: 81 (Fle_8_S1) and 83 (Rem_1_S2).

Particle amount

This paragraph presents the sensitivity analysis for the particle amount. The number of barite particles injected into the fractures varied between 0.5 and $6x10^6$ particles. The analysis presented in the previous paragraphs demonstrated that the fraction of deposited particles changed only slightly with higher particle numbers (Figure 28). Figure 29 A and B present the deposited volume fraction (in %), which compares the volume of deposited barites with the fracture volume (calculated as: $volume_{deposited\ barites}/volume_{fracture}$ x 100). This fraction was chosen since it allowed the comparison of both fractures and the different parameters. Furthermore, it allowed an estimation of the impact of the particle deposition. The absolute number of deposited particles, and therefore the deposited barite volume, rose linearly when more particles were injected. This behaviour occurred since the interaction of particles with other particles was not considered in the simulations. In reality, a deviation from this linear relationship is expected since the number of particles influences particle transport and deposition. In the Remlinger fracture, due to the smaller volume, the influence of the deposition (up to 0.2 %) was almost twice the value of the Flechtinger fracture (up to 0.12 %). The cumulative breakthrough curves (Figure 29 C, D) showed a similar trend (higher number of deposited particles, with a higher amount of injected particles) and highlighted that more particles (about one order of magnitude) were deposited in the Remlinger fracture. Furthermore, the shapes of the cumulative breakthrough curves demonstrated differences between the migration behaviour in the Flechtinger and Remlinger fractures: in the Flechtinger fracture, two widely spread peaks were observed. Whereas the particles in the Remlinger fracture were migrating with a similar breakthrough time through the fracture, resulting in one narrow peak with less tailing. That behaviour could be explained by faster mean velocities that can be expected in the Remlinger fracture (Darcy 1856, Figure 27), due to the constant flow rate (of 2 mL/s) and the smaller mean aperture, compared to the Flechtinger fracture. However, since the Remlinger fracture is smaller, more interactions of the particles with the fracture walls were expected, leading to a delay in the breakthrough times (see Figure 38 for a comparison of collisions of particles with the walls of the Flechtinger and Remlinger fracture). Furthermore, the time of the first breakthrough did not change with a higher particle amount.

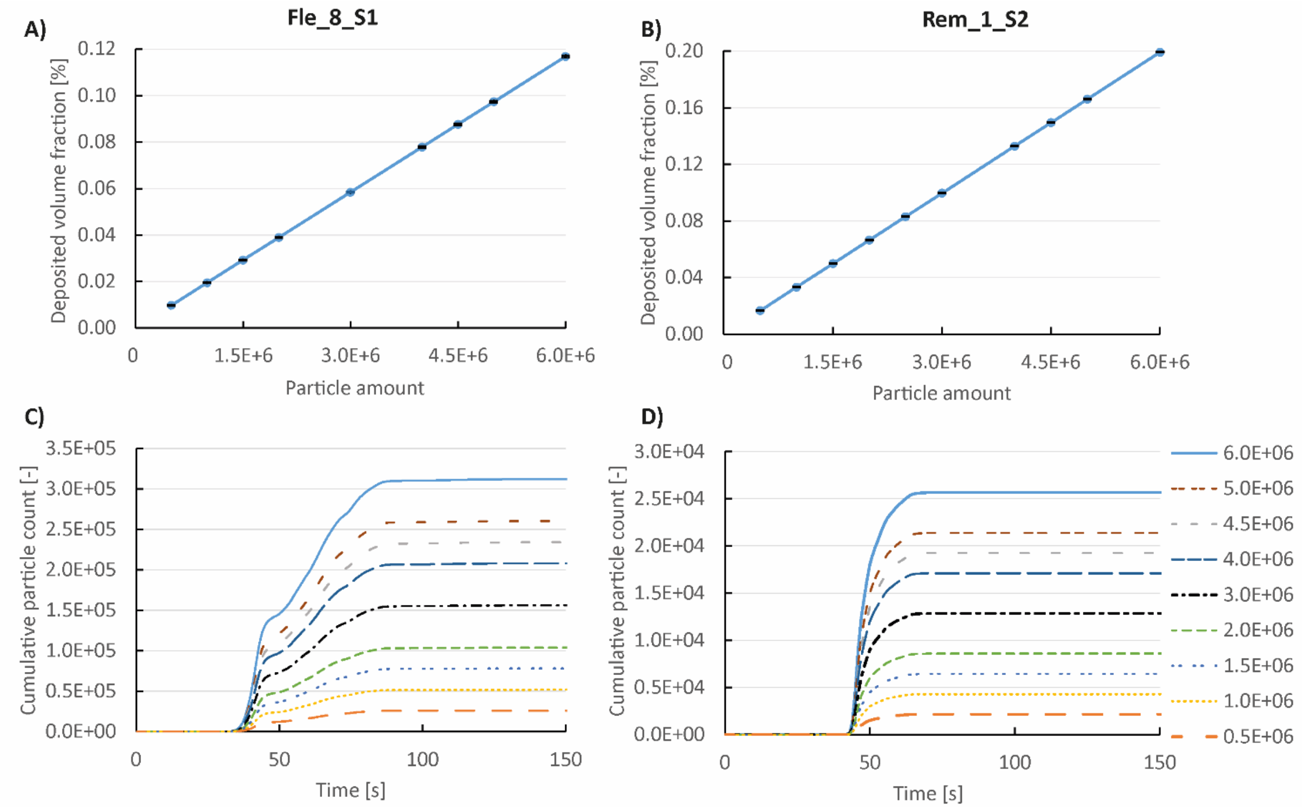


Figure 29. Varying particle amount
The volume of the deposited particles, compared to the fracture volume of the Fle_8_S1 ***(A)*** *and Rem_1_S2* ***(B)*** *fracture. A higher number of injected particles led to a linear increase in deposited volume. The 5 % error bars are shown for each result in (A) and (B). The cumulative breakthrough curves* ***(C, D)*** *showed a similar behaviour.*

Flow rate

Flow rates between 0.5 and 8 mL/s were simulated to investigate the influence of flow rate on hydro-mechanical particle deposition. For both fractures, less deposition with a higher flow rate was observed. This result agrees with literature data, where the decrease in deposition can be explained by a decreased residence time (lowering the probability of interactions of particles with the fracture walls) and higher hydrodynamic forces (e.g., Vilks et al. 2008). The residence time of each particle is defined as the time the corresponding particle remains in the structure until it escapes the structure or is caught inside. A decrease in mean residence time was observed for both fractures (Figure 31). However, the decrease in residence time with faster flow rates was small due to two reasons: (i) most particles were caught in the fracture, and (ii) especially with small flow rates, it was observed that most particles were caught in the first half of the fracture. With higher flow rates and, therefore, higher advective forces (see equations 12), more particles were transported further into the fracture. This can be explained by the increased transport velocities of the particles, which allowed the particles to escape the fracture surfaces instead of being caught. However, most particles were caught (Figure 32) since the increased travel distance also led to an increase in collisions with the fracture walls (Table 11), which decelerated the particles. Particles were caught on the surface when the deceleration by collisions and lower flow velocities decreased the transport velocity below the critical velocity of the Hamaker model (see equation 15).

Hence, the mean number of collisions that particles experienced while migrating through the fractures increased with higher flow rates. However, the increase in mean bounces became smaller with higher flow rates (Table 11), and the difference between the Flechtinger and Remlinger samples became smaller. Particles migrating through the Remlinger fracture experienced higher flow velocities but also a higher probability of collisions with the fracture walls due to the smaller apertures of the Rem_1_S2. The particles gained more energy by advective forces and could tolerate more collisions before being caught due to adhesion (see equations 12 & 15). However, due to the increased number of collisions, the combined effect was a greater delay in breakthrough time and more deposition. Additionally, lower residence times could be caused by the higher number of collisions and by the low number of particles that escaped the fracture (Figure 31).

The influence of the flow rate differed for the Flechtinger and Remlinger fractures. At the lowest flow rate, almost all barite particles were deposited in both fractures: 99.78 % in the Flechtinger sample and 99.99 % in the Remlinger sample. With the highest flow rate, 83.01 % of the particles were deposited in the Flechtinger fracture and 95.53 % (Figure 30 A, B) in the Remlinger fracture. The deposited volume fraction of the Rem_1_S2 was about 1.5 – 2 times greater than the fraction in the Fle_9_S2 due to a lower fracture volume and more deposition in the Remlinger fracture (Figure 30 C, D).

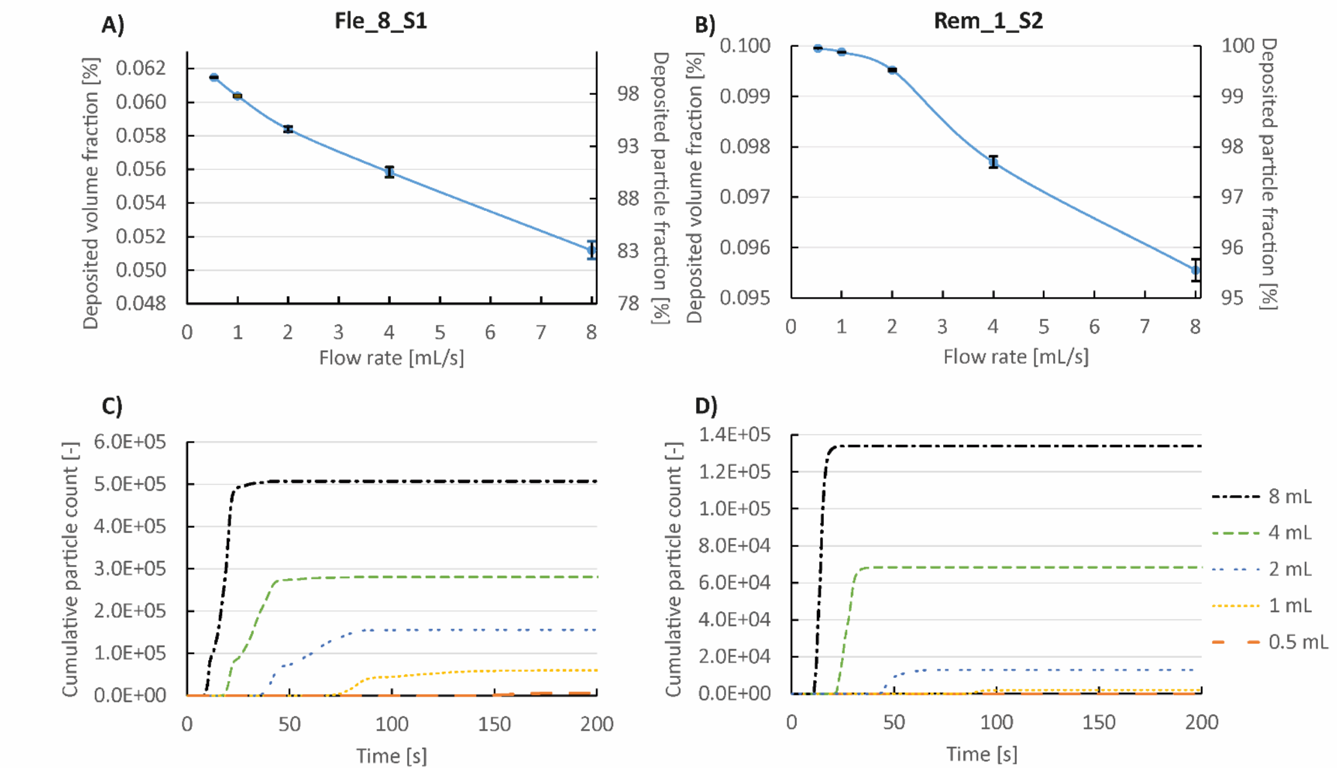


Figure 30. Varying flow rate
*Influence of the flow rate on particle deposition in the Flechtinger **(A)** and Remlinger **(B)** fracture. The cumulative breakthrough curves **(C & D)** show an earlier breakthrough and a higher number of escaped particles in both fractures. With a flow rate of 0.5 mL/s, only seven particles escaped the Rem_1_S2 and about 6000 the Fle_8_S1; hence, these two curves are marginally visible in the cumulative particle curves.*

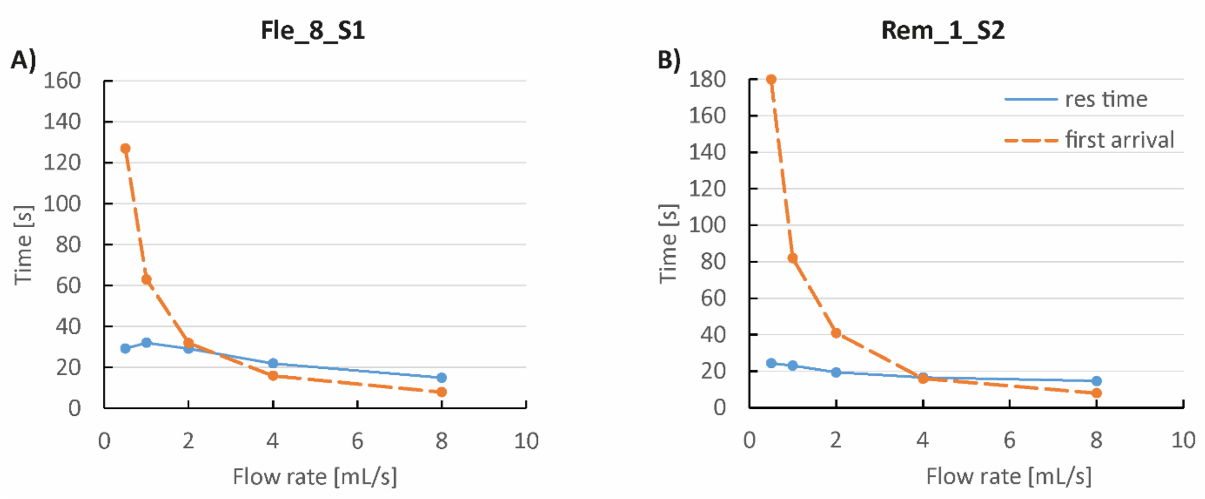


Figure 31. Residence time and first arrival with a varying flow rate
*First breakthrough times and mean residence times for the flow rate sensitivity analysis on the Flechtinger **(A)** and Remlinger **(B)** fractures*

Table 11. Mean particle – wall collisions in the reference samples
Comparison of the mean number of bounces/collisions that particles experienced in both fractures, with different flow rates

Flow rate [mL/s]	Mean number of bounces per particle Fle_8_S1	Mean number of bounces per particle Rem_1_S2	Ratio mean bounces Rem_1_S2/Fle_8_S1
0.5	34.65	94.77	2.74
1	79.23	171.1	2.16
2	145.8	249.8	1.71
4	218.5	339.6	1.55
8	285.5	420.4	1.47

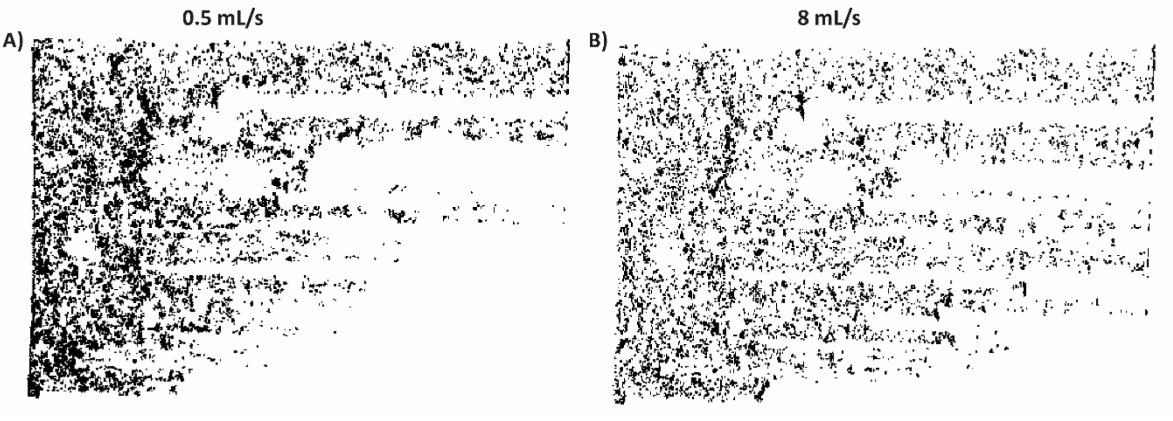


Figure 32. Influence of the flow rate on barite particle deposition
*For a low flow rate (0.5 mL/s, **A**), more particles are caught in the Fle_8_S1 fracture and earlier compared with higher flow rates (8 mL/s, **B**). For better visualization, the 10 µm particles were enlarged by a factor of 50, and the structure/flow fields were set as invisible.*

Temperature

The temperature was varied between 38 and 78 °C (while keeping the flow rate constant) to investigate the corresponding impact on the particle deposition. The temperature influenced the diffusion coefficient of the particles and the dynamic viscosity of water in the models. The former varied between 1.25 - 2.33x10^{-14} m²/s (calculated with equation 13 & 14), and the latter between 3.64 – 6.78 x10^{-4} kg/m*s (after Kestin et al. 1978). For both fractures, the simulations revealed only minor differences within the 5 % margin of error (Figure 33). That behaviour can be expected, since the transport for high Pe numbers (when Pe >> 1, for the applied particle sizes Pe between 10^4 – 10^6 were calculated with equation 16) is dominated by advection, and in these situations, the diffusive transport does not affect the particle migration or deposition (e.g., Dou et al. 2018, 2021). Although the temperature had a neglectable influence on the transport of existing particles, it is still an important factor to be considered in geothermal power plants. The temperature of the injected water depends on the extracted energy at the heat exchanger, and the temperature affects particle nucleation and growth (Tranter et al. 2021b). Hence, temperature changes also influence the amount and size of precipitated barite particles.

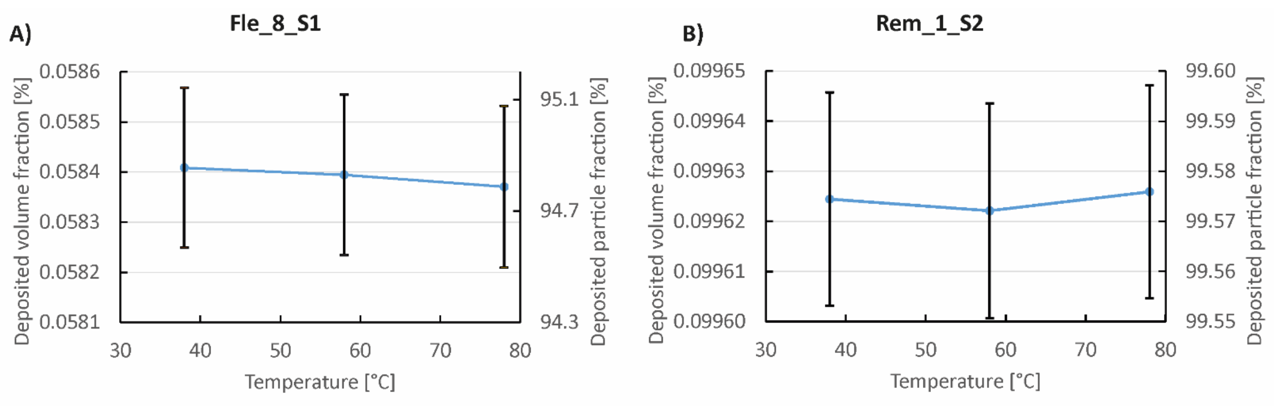


Figure 33. Varying temperature
Influence of the temperature on the barite particle deposition in the Flechtinger ***(A)*** *and Remlinger* ***(B)*** *fracture*

Hamaker constant (adhesion forces)

This paragraph presents the results from the sensitivity analysis of the Hamaker constant. Values between 10^{-24} – $5x10^{-22}$ J were applied to simulate different adhesion forces. When a particle was transported to the vicinity of a surface, it was caught when the particle was decelerated below a critical velocity (see equation 15). The Hamaker constant influenced the minimal velocity a particle must possess to be able to escape the adhesion forces from the rock surface. This critical velocity is presented in Figure 35 and ranges from about $4.2x10^{-5}$ to $9.5x10^{-4}$ m/s for the Hamaker constants considered in the sensitivity analysis. With a higher critical velocity (due to a higher Hamaker constant), particles were caught earlier in the structure, and the number of deposited particles increased (Figure 34). This was also reflected in the deposited volume fraction calculated for both fractures (Figure 36 A, B). In the Fle_8_S1, about 85 – 99 % of the particles were deposited, resulting in a deposited volume fraction of 0.053 – 0.061 %. More particles were deposited in the smaller Rem_1_S2 (96 – 100 %), which resulted in a higher deposited volume fraction of 0.096 – 0.1 %. For both fractures, the Hamaker constant had a large influence on the deposition: in the Flechtinger sample, the number of escaped particles ranged from 348 – 421534 (for the highest to lowest Hamaker constant) and for the Remlinger sample from 0 – 109656 (Figure 36 C, D). In general, the first breakthrough time remained constant when the Hamaker constant was varied. Two exceptions were observed in the Rem_1_S2 fracture, for the highest (0 particles escaped the fracture hence no breakthrough was observed) and the lowest Hamaker constant. In the latter case, the first breakthrough of about 60 particles was detected 10 seconds earlier compared to the other simulations. This effect could be explained by a faster flow path, where at some point, the particles were decelerated between the critical velocities of the two lowest Hamaker constants (4.21 – 9.42x10^{-5} m/s). Thus, this flow path only became accessible when the lowest Hamaker constant was applied, and this deceleration was not sufficient to catch the particles.

The overall lower deposition in the Flechtinger sample can be explained by the lower amount of collisions a particle experienced while migrating through the fracture: in the base model of the particle migration, a particle that was transported through the Rem_1_S2 collided about 1.7 times more with the fracture walls (Table 11), which resulted in a greater loss of energy and a higher likelihood of deposition. This effect was partly compensated by the higher flow rates in the Remlinger fracture.

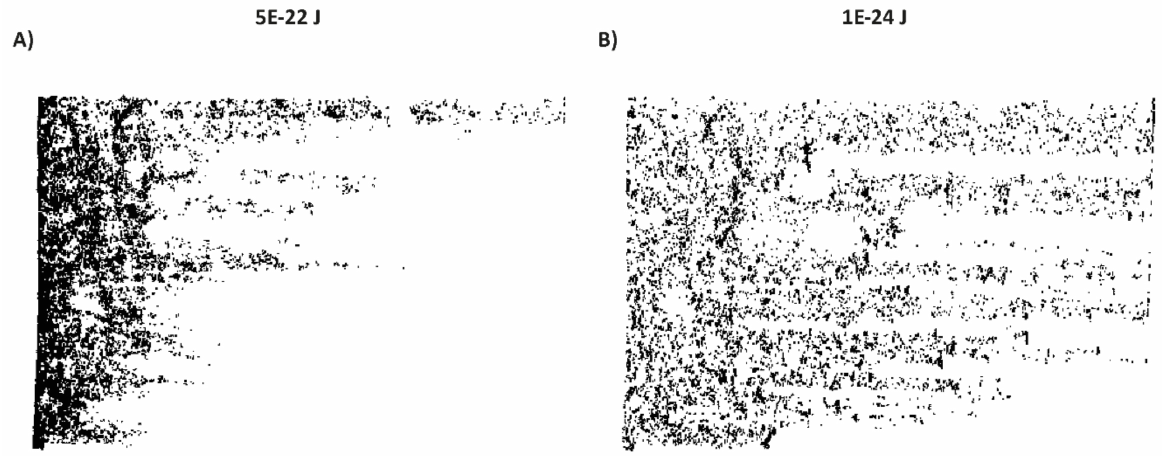


Figure 34. Illustration of particle deposition with varying adhesion forces
Particle deposition pattern for the highest ***(A)*** *and lowest* ***(B)*** *Hamaker constant applied on the Fle_8_S1*

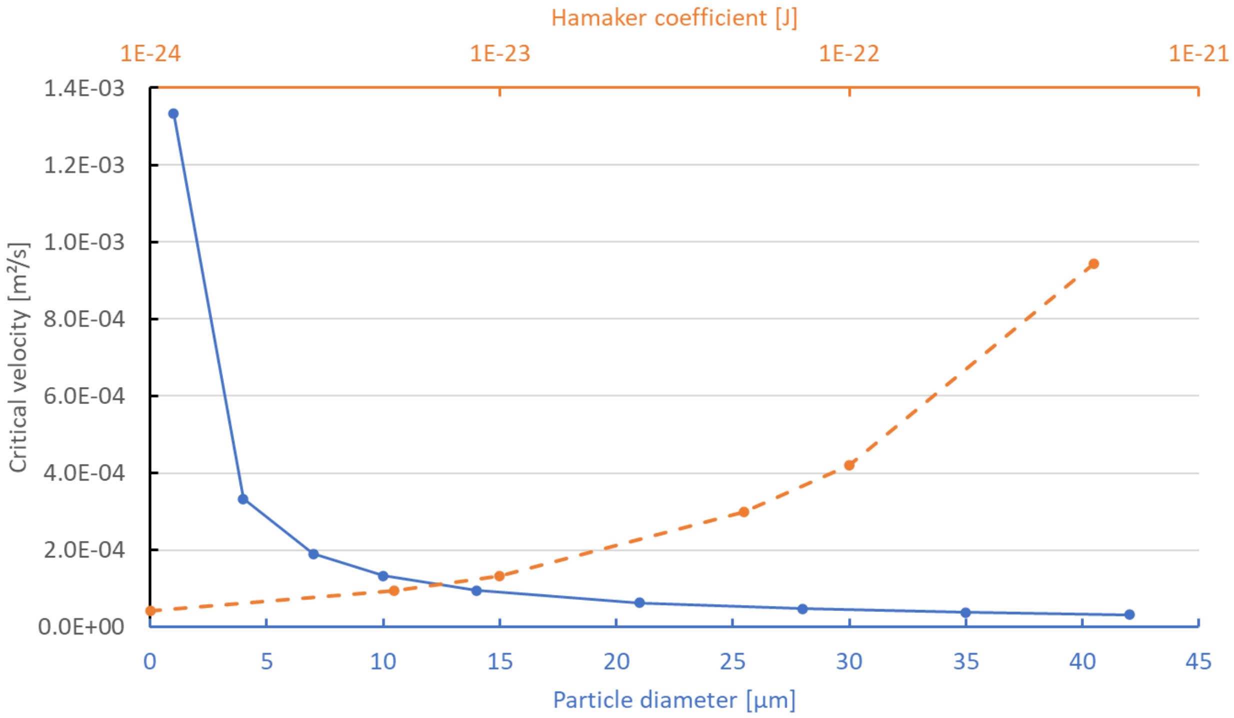


Figure 35. Critical velocity with varying parameters
The critical velocity of the Hamaker collision model depends on the Hamaker constant and particle size. When a particle was transported in the vicinity of the rock surface (4x10^{-10} *m in the simulations), the critical velocity determined if the particle was caught or not. Faster particles escaped the adhesion forces. The critical velocities were determined according to equation 15. For the varying Hamaker constant (the dashed orange curve, distances plotted at the secondary X-axis are logarithmically spaced), the particle size was fixed at 10 µm; for the varying particle size, the Hamaker constant was fixed at* 10^{-23} *J.*

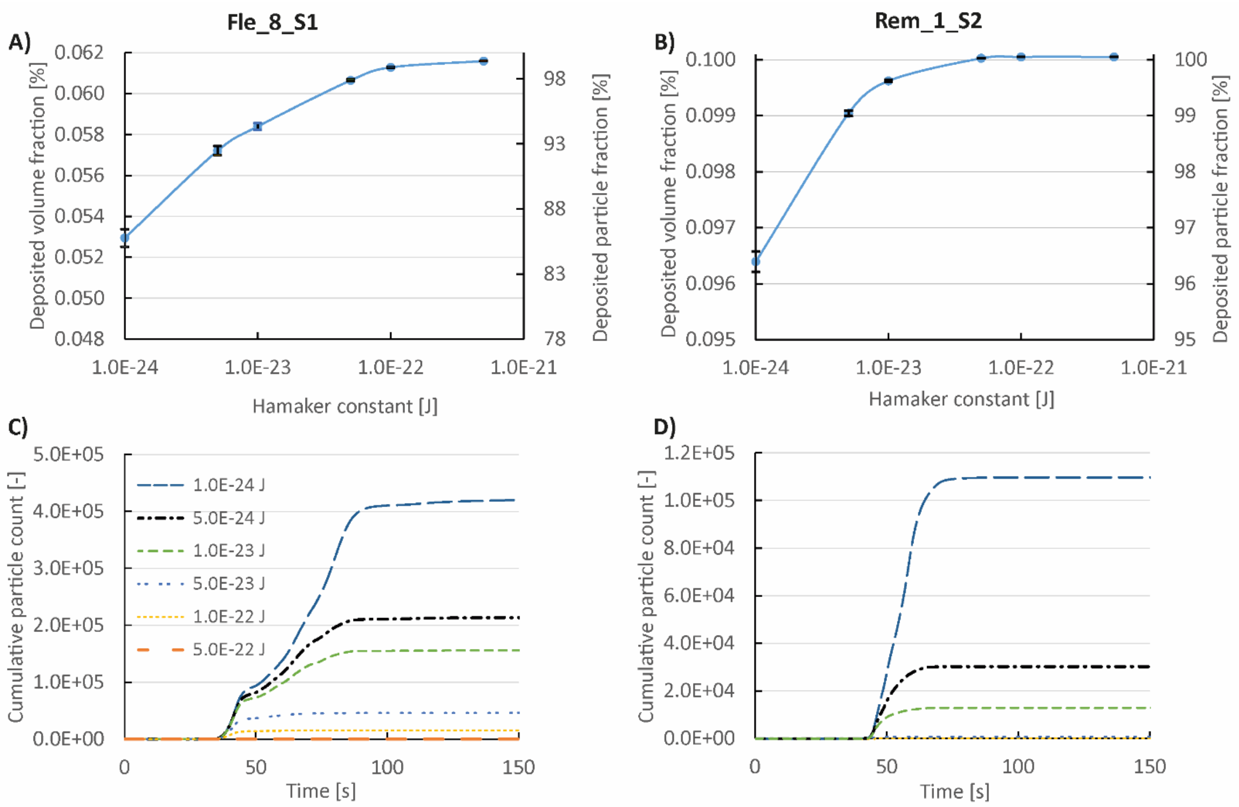


Figure 36. Varying adhesion forces

Influence of the Hamaker constant on particle deposition in the Fle_8_S1 ***(A)*** *and Rem_1_S2* ***(B)****. Note: the distances plotted in the X-axis are logarithmically spread.* ***(C, D)*** *show the cumulative particle counts for both fractures.*

Particle size

A particle diameter of 1 to 42 µm was applied for the sensitivity analysis presented in the following paragraphs. Larger particles resulted in a higher deposited volume fraction in both fractures, ranging from 0.000062 – 3.35 % for the Fle_8_S1 and 0.0001 – 6.67 % for the Rem_1_S2 (Figure 37 A, B). The amount of injected particles was constant for all tests, and due to the varying particle size, the total volume of the injected particles increased with the particle size. Thus, the influence of one deposited particle on the deposited volume fraction depended on the particle size. Another observed effect was the decrease in the fraction of deposited particles with greater particle sizes (100 – 74 % for the Fle_8_S1 and 100 – 87 % for the Rem_1_S2). This can be explained with the Hamaker collision model, where an increased particle size results in a decreasing critical velocity (Figure 35). Thus, bigger particles can experience more deceleration than smaller particles before being caught by the adhesion forces. Although bigger particles also had a higher probability of interacting with the fracture walls, the overall effect of rising particle size was a higher probability of escaping the fracture. Generally, fewer particles were deposited when particle size increased (Figure 37 C, D), but due to the greater volume of each particle, the deposited volume fraction increased. The first breakthrough time was constant for all simulations conducted on the Flechtinger sample. However, a faster breakthrough was detected in the Remlinger fracture, for the bigger particle diameters (≥ 21 µm, Figure 37 D). An earlier breakthrough could be explained by the size exclusion effect (surface charges were neglected in the simulations; Zhang et al. 2012) and faster flow paths that only support the transport of bigger particles since smaller particles might be deposited along these streamlines (due to the deceleration between the critical velocity of 14 and 21 µm particles, at one or multiple spots).

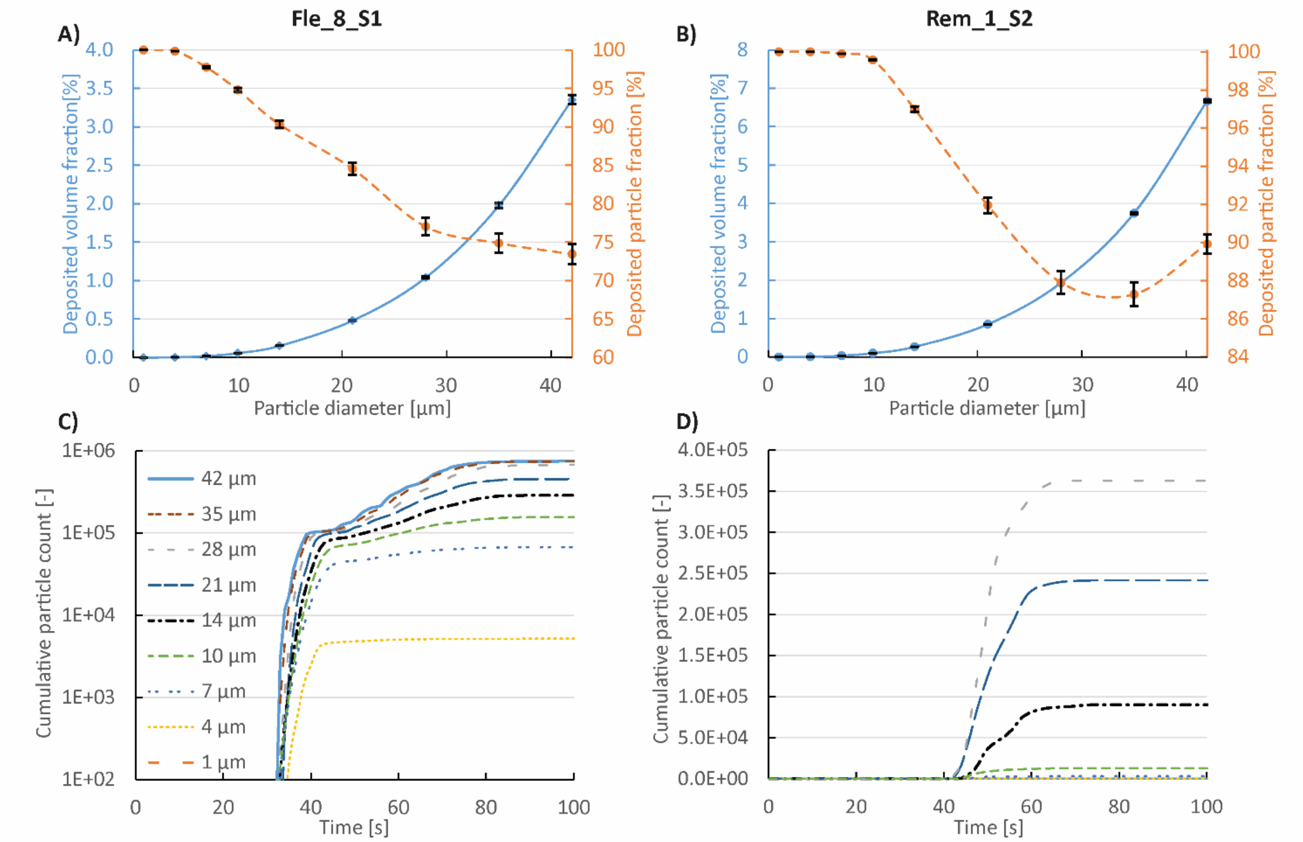


Figure 37. Sensitivity analysis for the particle size
The deposited volume fraction and fraction of deposited particles for both fractures are presented in ***(A)*** *and* ***(B)****. The cumulative particle count shows the breakthrough behaviour for each particle size simulation in the Fle_8_S1* ***(C)*** *and Rem_1_S2* ***(D)****. Note: some curves are not shown due to the low amount of particles that escaped the structure. The simulations of the two biggest particle sizes migrating through the Remlinger fracture were only conducted with* $1x10^6$ *particles since a higher amount resulted in too many collisions that could not be simulated with the available computational resources. Therefore, both curves are not shown in (D). In order to include the results from the biggest particle sizes in the plot (B), the deposited particle amount was multiplied by three since the relationship between particle amount and the number of deposited particles is approximately linear (see the paragraph "Particle amount" in subsection 4.3.2).*

When the particle size reaches a similar range (or becomes larger) to small aperture areas, some flow paths might not allow these particles to pass through the structure (similar to the size exclusion effect in porous media, e.g., Cumbie and McKay 1999; Molnar et al. 2019). With bigger particles, more small aperture zones might prevent the particles from passing through, resulting in an increase in the deposited particle fraction. Additionally, with larger particles, the probability of collisions with the fracture walls increased (Figure 38). Since collisions reduce the particle velocity, increasing collisions favour an increasing particle deposition. Thus, the Hamaker collision model and the physical effects of increasing the particle size can explain the observed particle deposition behaviour. With an increasing particle size, an initial decrease in particle deposition was attributed to the sharp decrease of the critical velocity in the Hamaker model. The influence of the increasing number of collisions rose with larger diameters, resulting in a smaller decrease in particle deposition until a plateau was reached (for a particle diameter of 28 – 42 µm in the Flechtinger fracture and particle diameters of 28 – 35 µm in the Remlinger fracture). With larger particle diameters, the deposition increased, as observed for the 42 µm particles in the Remlinger fracture. Since the aperture distribution has a large influence on the interaction with particles, this effect was observed earlier (with smaller particles) in the smaller Remlinger fracture (with a hydraulic aperture of 93.66 µm) than in the bigger Flechtinger fracture (also see Figure 24). Compared with the Flechtinger fracture, more collisions occurred in the Remlinger fracture at a given particle size, and the increase in the number of collisions with particle size was about 70 % greater. This difference can also explain the higher particle deposition in the Remlinger sample. Furthermore, with the available computational resources, it was not feasible to simulate the high number of collisions that the biggest particles (35 and 42 µm) experienced in the Remlinger fracture. Therefore, these two simulations had to be carried out with a reduced number of particles ($1x10^6$). The number of deposited particles was multiplied by a factor of 3 to compare the results with the results from the other simulations (a linear relationship between particle amount and amount of deposited particles was demonstrated in subsection 4.3.2.).

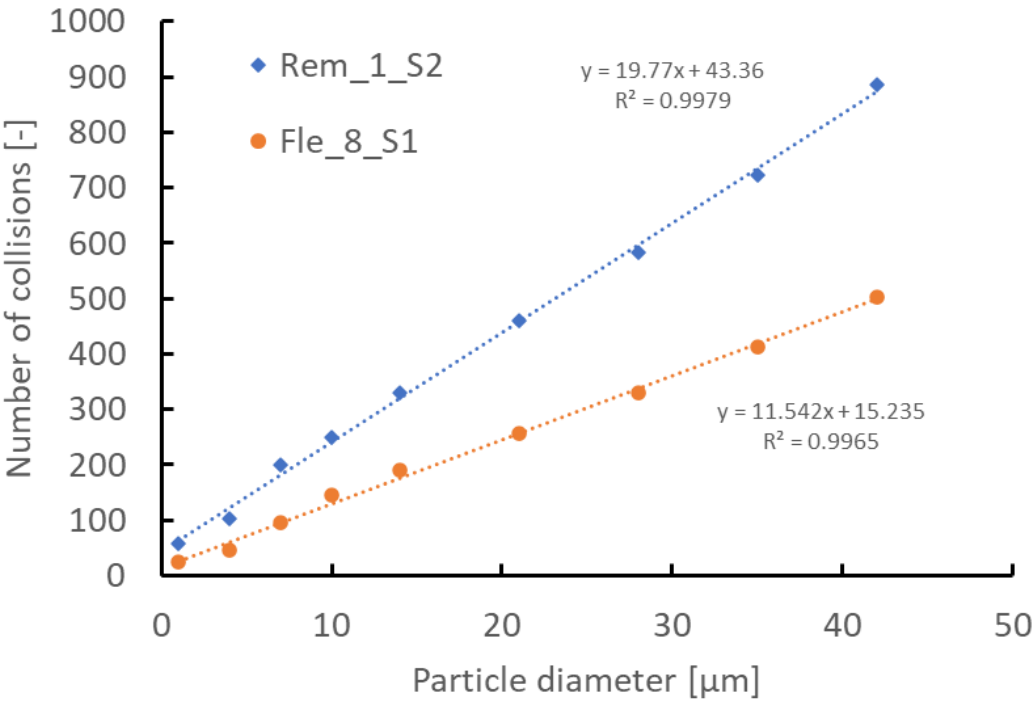


Figure 38. Number of collisions for each particle size
Particles migrating through the smaller Remlinger fracture collided more frequently with the fracture walls, with a greater increase with increasing particle size. Linear fitting curves with a slope of about 19.8 (Rem_1_S2, R^2 = 0.9979) and 11.5 (Fle_8_S1, R^2 = 0.9965) were calculated.

The particle size sensitivity analysis was additionally conducted with a constant barite mass to compare the deposition when a fixed mass of particles (in varying sizes) is injected into the fractures. Therefore, for each particle size (diameters ranging from 4 to 42 µm), a different amount was injected to keep the injected barite mass at a constant value of 1 mg. The results for both fractures are presented in Figure 39 A and B. Since most small particles were caught in both fractures, the deposited volume fraction had the highest values for the smallest particle size. As elaborated in the previous paragraphs, with greater particle diameters, more particles escaped the fracture until a plateau was reached. With even bigger particles, the number of escaped particles declined again. This suggests that an optimal particle diameter exists, where the deposition reaches a minimum. Since the physical interactions of particles with the fracture wall increase with an increasing diameter, this optimum diameter depends on the aperture distribution of the fracture. For the Flechtinger fracture, the optimum particle size could be reached between 28 – 42 µm but could also include larger particles that were not simulated in this study. For the smaller Remlinger fracture, the optimal particle diameter was between 28 – 35 µm, with the lowest fraction of deposited particles and the lowest deposited volume fraction. The idea of an optimal particle size for particle transport through fractures can also be found in the literature (e.g., Cumbie and McKay 1999).

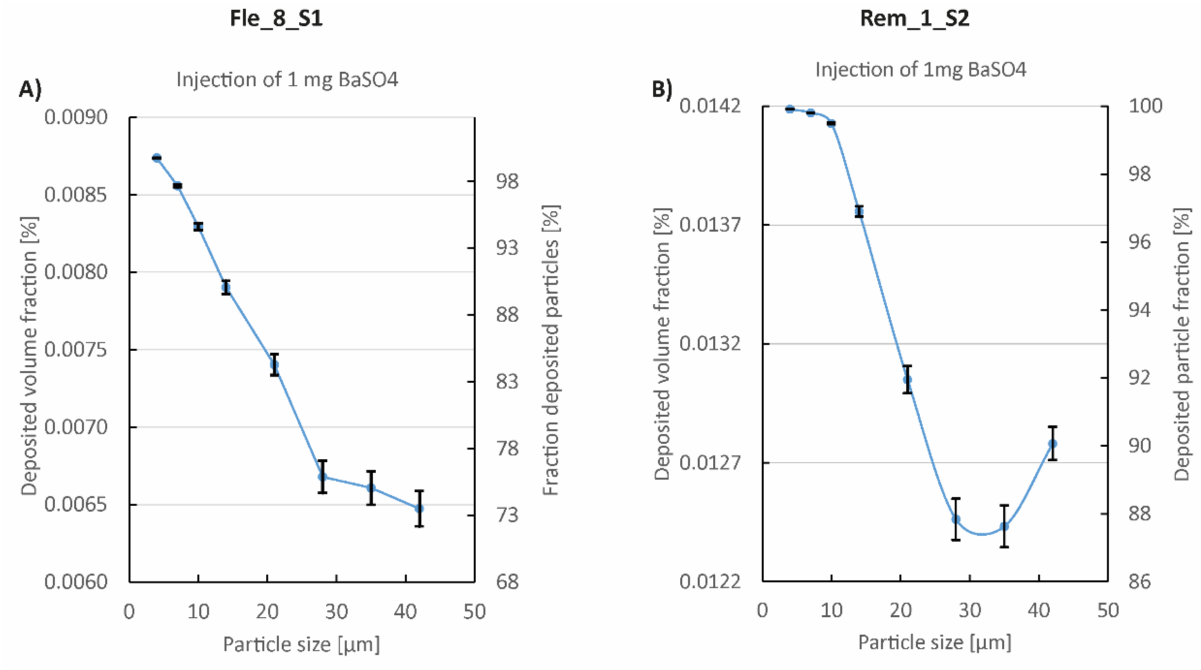


Figure 39. Simulation results of injecting a constant mass of barite into the fractures
For each of the eight batches, 1 mg of barite was injected as particles with a constant diameter, which was different for each batch. Thus, the combined effect of varying particle size and particle amount was considered in the simulations. ***(A)*** *shows the result for the Flechtinger sample and* ***(B)*** *the result for the Remlinger sample*

Comparison with tracer particles

This paragraph of the sensitivity analysis is dedicated to the comparison of the transport of barite particles and a conservative tracer under the same flow conditions (identical to the parameters of the base model presented in Table 10). In order to simulate a conservative tracer, water molecules were selected (also see section 4.2), with a diffusion coefficient of 4.58×10^{-9} m²/s (the diffusion coefficient of water at 58°C was estimated after Holz et al. 2000). Collisions of the tracer molecules with the fracture walls were simulated with the sieving model (where the restitution value was set to 1). A random seed of 81 was used for the placing of the molecules and particles. Due to computational limitations (owing to the high number of collisions that must be computed), the number of particles and molecules was reduced to 1×10^5.

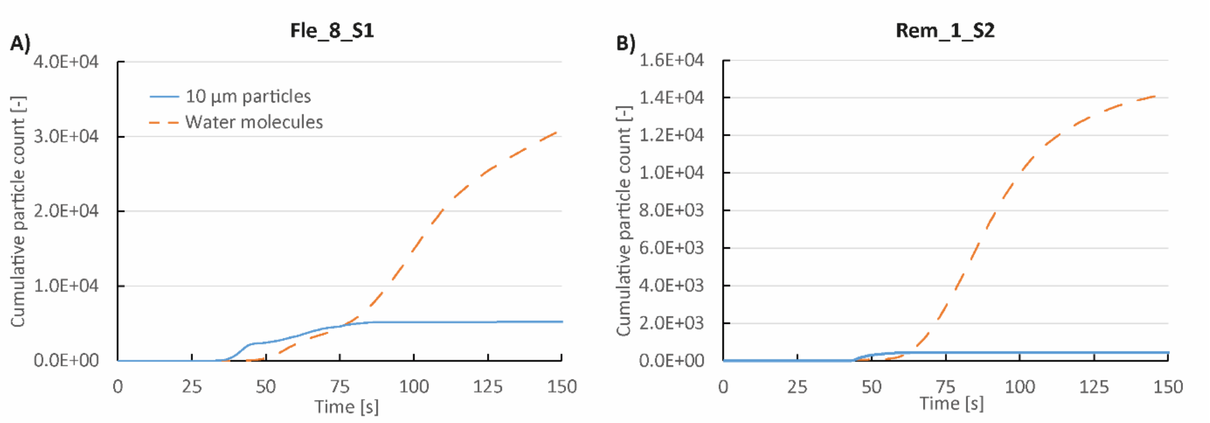


Figure 40. Comparison of particle and tracer transport
Comparison of the transport of barite particles (10 µm diameter) and a conservative tracer (water molecules) through the Flechtinger ***(A)*** *and Remlinger* ***(B)*** *fracture.*

For both fractures, the same behaviour was observed (Figure 40). The tracer particles had a 30 % later breakthrough time, a higher recovery rate, and a distinctive tailing compared to the barite particles. Furthermore, the mean number of collisions a molecule experienced was about 30 times higher than for barite particles, which can be explained by the higher diffusion coefficient of tracer molecules (the diffusion coefficient of the tracer molecules was about five orders of magnitudes higher compared to the barite particles, with respective values of about $4x10^{-9}$ and $2x10^{-14}$ m^2/s), which led to a higher deviation from the flow paths and therefore a higher probability to collide with the fracture walls. Furthermore, the probability that tracer molecules were caught in the structure was lower due to the different collision model (sieving) and the higher mobility of the tracer molecules (due to the higher diffusion coefficient). Therefore, the molecules were able to travel longer than the barite particles, and more molecules reached the outflow. Since molecules experience more diverse transport paths, a pronounced tailing was observed in the breakthrough curves. Particles and molecules collided more frequently with the Remlinger fracture walls than with the Flechtinger fracture walls. The mean collisions of about 250 (particles) and 7776 (molecules) were simulated in the smaller Remlinger fracture, and mean collisions of about 145 (particles) and 4939 (molecules) were calculated in the Flechtinger fracture.

The greater influence of Brownian motion on the tracer transport resulted in a greater deviation from the faster main flow paths (higher Taylor dispersion), a higher probability of reaching stagnant zones with a low flow velocity, and more collisions with the fracture walls (Becker et al. 1999; Mortensen 2001; Zvikelsky and Weisbrod 2006). Particles, on the other hand, tend to stay on the same streamlines, which results in an enhanced transport velocity and an earlier arrival (Knapp et al. 2000).

Since the diffusion coefficient is size dependent, this effect is greater for bigger particles (Chrysikopoulos and Abdel-Salam 1997; also see equation 13 & 14). Other effects on the enhanced transport of particles (or colloids) in real fractures include the size exclusion effect, the charge exclusion effect, or matrix diffusion (Oswald and Ibaraki 2001; Zhang et al. 2012). The former considers the restricted transport paths due to the greater size of particles (i.e., particles cannot enter smaller pores of the rock matrix or small aperture zones), and the latter considers the repulsion of negatively charged particles (at neutral pH) from the negatively charged rock surface (Zhang et al. 2012). Since the electrostatic forces were neglected in the simulations (see section 2.6), the charge exclusion effect does not have an influence on the presented results. The matrix was simulated as an impermeable material due to the low permeability measured in the laboratory and reported in the literature (see section 3.1). Depending on the host rock and the size of the suspended solids, matrix diffusion can influence the migration of colloids or particles (Oswald and Ibaraki 2001). The size exclusion effect could become important for the bigger particles simulated in this study.

Particle deposition and flow velocity

Barite particles were preferentially deposited in low-flow velocity areas (Figure 41). Due to the low diffusion coefficient, the particle transport velocity was dominated by advection (with Pe of $10^4 – 10^6$). Hence, with low flow velocities, the particle velocity was also greatly reduced. This increased the probability of decelerating particles below the critical velocity of the Hamaker model. The particle velocity was further decreased after collisions with the fracture walls, which happened more frequently with the higher residence times in the low-flow velocity voxels (Stoll et al. 2019). With an increased particle diameter, the critical transport velocity decreased (Figure 35), i.e., bigger particles escaped the adhesion forces even with smaller transport velocities compared to smaller particles. Thus, big particles were only deposited when the fluid flow reached very low flow velocities (Figure 42). The flow velocities in voxels with deposited particles revealed that most particles in the Remlinger fracture were deposited when the flow velocity dropped below 10^{-3} m/s. About 90 % of the injected particles were deposited when the flow velocity was below $5x10^{-5}$ (for the 42 µm particles), $6x10^{-5}$ (for the 35 µm particles), $1x10^{-4}$ (for the 28 µm particles), $2x10^{-4}$ (for the 21 µm particles), $5x10^{-4}$ (for the 14 µm particles) and $1x10^{-3}$ (for the 10 µm particles) m/s.

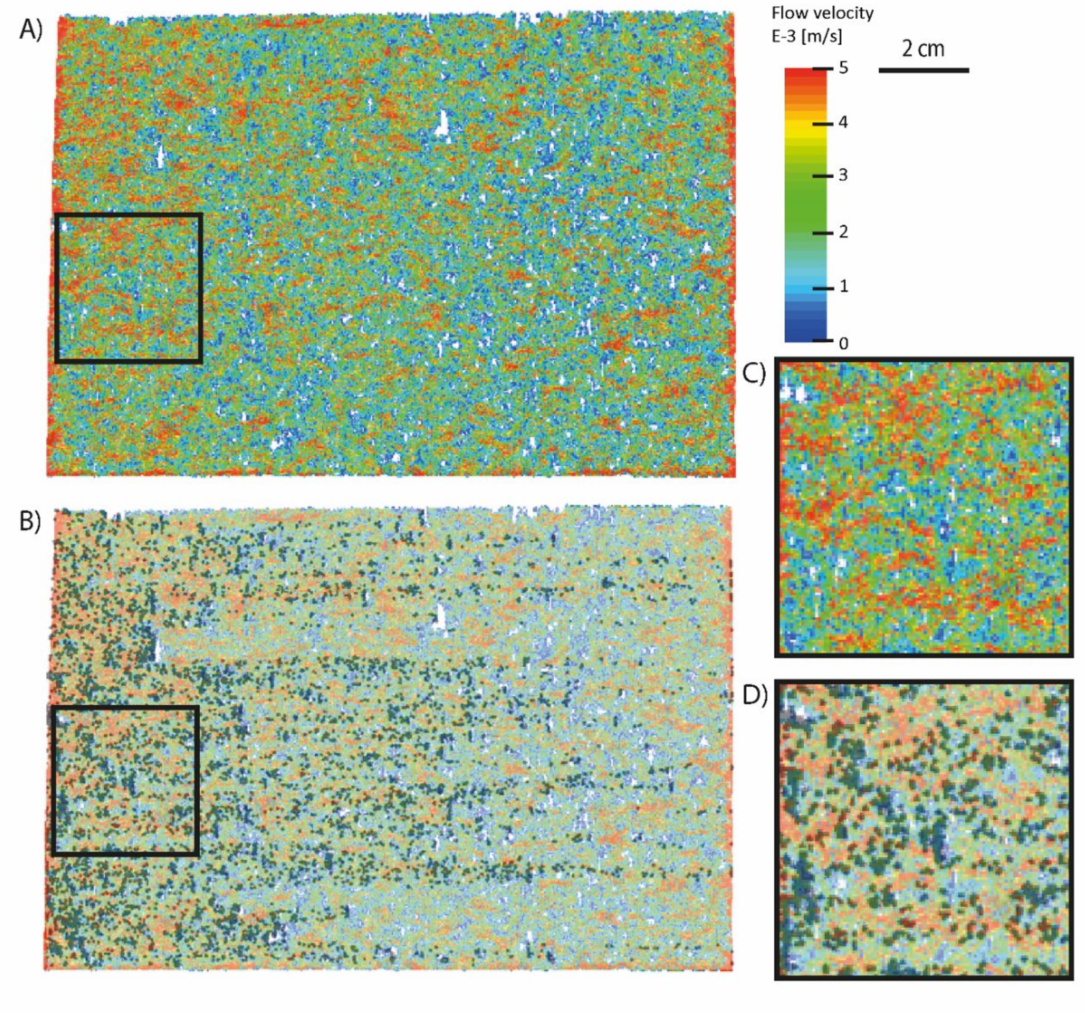


Figure 41. Particle deposition pattern
*The flow field **(A)** for the base model in the Remlinger fracture was computed with a pressure drop of -520 Pa, which resulted in a flow rate of 2 mL/s. **(B)** The deposited barite particles with a diameter of 14 µm (for the visualization, the size was enhanced by a factor of 50) with the flow field in the background. The close-up of the flow field **(C)** and particle deposition **(D)** show that particles preferentially deposit in low-velocity zones. A similar behaviour was observed for the Flechtinger fracture.*

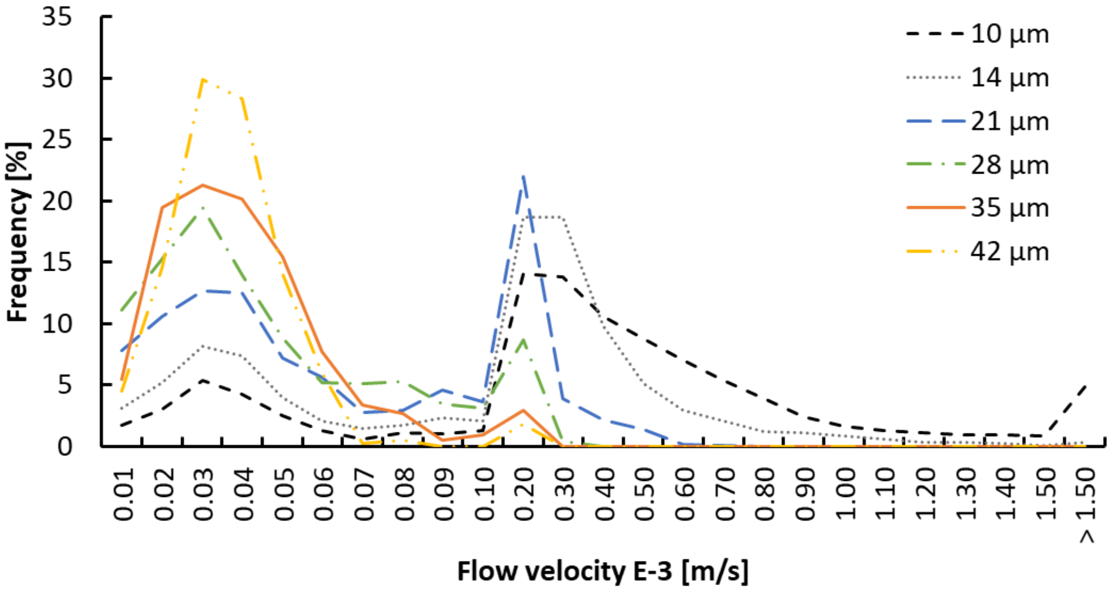


Figure 42. Flow velocity and particle deposition
Histogram of the flow velocities in voxels, where particles were deposited in the Remlinger fracture. Larger particles were only deposited in voxels with very low flow velocities. With smaller diameters, the particles were also deposited in voxels with a faster flow velocity. Most particles were deposited when the flow velocity was below 10^{-3} m/s.

4.3.3 Discussion and conclusion

Fracture sealing due to barite scaling was described theoretically in laboratory experiments and observed at multiple geothermal sites, influencing the productivity of these power plants (Nitschke et al. 2014; Regenspurg et al. 2015; Griffiths et al. 2016; Orywall et al. 2017). The corresponding sealing rates were reported to be on the order of days to months, where the dependence on the mean fracture aperture was quantified by Tranter et al. (2021b). In their modelling study, scaling was induced by injecting a supersaturated solution, i.e., barite nucleation and growth were considered. The fast sealing of their smallest analysed fracture (100 µm mean aperture) showed that a critical amount of barite can quickly accumulate and significantly influence fracture permeability. Alternatively, these scaling minerals can also form near the surface (e.g., in pipes, at the heat exchanger, or in solution), be transported into the reservoir, and therefore have the potential to deposit and influence the fracture permeability in a similar manner (Frick et al. 2011; Tranter et al. 2021b). This section investigated under which conditions these precipitated barite particles deposit in a single fracture. The base model was qualitatively validated with experimental data. A sensitivity analysis was conducted on two fractures to investigate the influence of temperature, flow rate, particle size, adhesion forces, and particle amount. The conditions in the experiments were considered in the range of the parameters.

An estimation of the numerical error demonstrated a small influence of the random seed and the particle amount on the fraction of deposited particles. Thus, the numerical error of the results was assumed to equate to the 5 % error bound from the flow field calculation. The results showed the largest influence on the hydro-mechanical deposition for the flow rate and particle size. With faster flow rates, the fraction of deposited particles decreased from about 100 % to 83 % (Flechtinger fracture) and 95 % (Remlinger fracture). Due to greater transport velocities (due to higher advective forces), fewer particles were trapped by adhesion. An increase in particle size decreased the fraction of deposited particles from about 100 % to 74 % (Flechtinger fracture) and 86 % (Remlinger fracture). Greater particle sizes decreased the critical velocity of the Hamaker adhesion model, i.e., bigger particles must be further decelerated to get trapped. However, due to particle sizes in the order of the fracture apertures, an optimal particle size was obtained for each fracture. For the Flechtinger fracture, the lowest deposition occurred for 35 – 42 µm particles and for the Remlinger fracture for 28 – 35 µm particles. Bigger particles had a higher chance of colliding with the fracture walls, leading to a higher number of collisions with increasing particle diameter. Since collisions reduce the transport velocity, this effect influenced the particle deposition. Another possible influence could be physical straining, which restricts the possible transport path for larger particles. For the biggest particle size (42 µm) in the Remlinger fracture, the high amount of collisions (and potentially the physical straining) was not compensated by the lower critical velocity and thus resulted in increased particle deposition. The optimal particle size was also obtained by simulating an injection of a constant mass (1 mg) of barite particles, where the combined effect of varying particle size and particle amount was considered. Generally, the fraction of deposited particles was greater in the Remlinger fracture, which can be explained by the smaller apertures, resulting in a higher probability of collisions with the fracture walls. An analysis of flow velocities demonstrated that particles were preferentially deposited in low flow velocity areas. Varying temperatures had a neglectable influence on the hydro-mechanical particle deposition due to the low diffusion coefficients and comparably high flow velocities. A decrease in adhesion forces led to a decrease in the fraction of deposited particles from about 99 to 85 % (Flechtinger) and 100 to 96 % (Remlinger).

The presented models allow some qualitative conclusions for selecting favourable conditions to minimize hydro-mechanical barite particle deposition in single fractured rocks: (i) higher flow rates, (ii) lower adhesion forces, (iii) lower particle amount, (iv) larger fracture apertures and (v) bigger particles, where the optimum size depends on the aperture distribution of the fracture. Furthermore, it was demonstrated that most deposition occurs in low-flow velocity areas. For more quantitative analysis, the following factors could be considered:

(i) Additional forces might influence the particle deposition, e.g., particle and surface charges or the gravitational force. For vertical fractures, the gravitational force accelerates or decelerates the particle transport (depending on the flow direction), which would have an influence on particle velocity and, thus, the deposition by adhesion forces. When the fracture orientation deviates from the vertical position, the gravitational settling could become an important deposition mechanism for bigger particles (Ding et al. 2021). Furthermore, the Hamaker coefficient should be determined experimentally to include a realistic value in the simulations.

(ii) Several complexities could be considered for the particles, e.g., particle growth, particle-particle interactions, or more realistic shapes. Alternatively, spherical particles (artificial or natural) could be used in the experiments to allow the direct implementation of similar particles in the simulations. Additionally, the migration of other relevant particles should be investigated (e.g., clay particles).

(iii) For this study, fracture geometries were reconstructed from CT-images. Thus, the isolated influence of different surface roughness values and aperture distributions were not considered. It is well known that roughness and the aperture distribution influence particle migration in fractures (e.g., James and Chrysikopoulos 1999; Shen et al. 2012; Stoll et al. 2019). Consequently, the presented study could be extended by introducing different roughness values (e.g., a sub-voxel roughness). The reconstruction method should be investigated in a benchmark study to quantify the reconstruction's uncertainties and to find optimal parameters for the workflow (as the optimal growing limit or the appropriate CT_{mat} value (local/global)). As elaborated in subsection 4.2.4, suitable samples should be selected for such a benchmark study, i.e., the ratio of fracture apertures to the reconstructed (resampled) spatial resolution should be as big as possible. The DTs can also differ from the real samples due to errors in the segmentation or CT measurement. Additionally, a fracture with a simple geometry (straight and smooth, e.g., 3-D printed) could be analysed as a reference sample through the whole DRP workflow to validate the results obtained after each step.

(iv) Since particle recovery rates can be highly variable (and depend on e.g., particle size and fracture aperture, e.g., Vilks and Bachinski 1996; Alaskar et al. 2015), a quantitative analysis of experimental data could provide valuable insights into the validity of the modelling results and the chosen parameters and simplifications. A defined number of (artificial) particles could be injected into the cores without recirculating the particle-fluid suspension. The deposited particles could be analysed by SEM or by preparing particles that illuminate or have a distinctive colour. In that way, the recovery rates and the deposition zones in the fractures could be identified and compared with simulations.

(v) The release of deposited particles could be included in the models to investigate the importance of this process (e.g., see Zhang et al. 2012; Yosri et al. 2021).

5. Summary and Outlook

This thesis presented the application of the DRP workflow to model barite particle migration in single fractured rocks. A framework was developed to create validated DTs at higher spatial resolution. This included finding the optimal segmentation method, developing a novel workflow for correcting the PVE, and discussing suitable validation parameters. A sensitivity study was conducted to investigate which parameters have the largest influence on the deposition of barite particles in fractures. The key findings and suggestions for future studies are summarized in the following paragraphs.

The first part of the analysis demonstrated the benefits of novel machine learning approaches during the segmentation procedure. The applied conventional methods were influenced by noise, ring artefacts, and operator bias. Additionally, a visual comparison revealed various errors in these segmentations. In contrast, both machine learning methods (i.e., CNN and random forest algorithm) resulted in superior segmentations without being influenced by ring artefacts or noise. However, current shortcomings of these approaches were discussed: the CNN method requires careful optimization of the network structure (and hyper parameters), which has high computational demands and can be time-consuming. Moreover, preparing an appropriate ground truth is challenging and can suffer from the operator bias. On the other hand, the RF is still influenced by an operator bias (because of the interactive training). In general, it was demonstrated that the RF segmentation provided reliable results and was the most suitable method for segmenting the fractured cores. However, the CNN approach has the potential to surpass the performance of the RF methods if current limitations can be overcome. A key advancement would be a comprehensive database with a sufficient amount of high-quality training data and corresponding ground truths. A comparison with laboratory data revealed large deviations of the properties calculated for the segmented fracture. Since the fracture apertures were in a similar range as the spatial resolution, these deviations were attributed to the PVE.

A novel workflow was presented to correct the influence of the PVE. The fractures were reconstructed at a higher resolution based on local aperture calculations with the CFMA method. The permeabilities and hydraulic apertures of all reconstructed fractures were validated with laboratory data. A comparison to the segmented fractures highlighted the large deviations of these structures and the significant improvement after applying the reconstruction workflow. Although for most reconstructed DTs, similar tracer breakthrough curves were simulated, larger deviations were revealed for most Remlinger fractures. This might be attributed to the limitation in enhancing the spatial resolution. The resampled spatial resolution was still in a similar range as the fracture apertures, leading to deviations from the real aperture fields and, thus, deviations in the flow fields. Since the Remlinger fractures were smaller than the Flechtinger fractures, this effect might have a larger influence on the Remlinger fractures and, thus, explain the deviations in the characteristics of the breakthrough curves. The large-scale parameters (i.e., permeability, mean aperture) were less affected by these deviations, thus were similar to laboratory data.

A sensitivity study was conducted to investigate the influence of various parameters (temperature, flow rate, adhesion forces, particle size, and amount) on the hydro-mechanical deposition of barite particles in single fractured rocks. The lowest deposition was determined for (i) higher flow rates, (ii) lower adhesion forces, (iii) lower particle amount, (iv) larger fracture apertures, and (v) bigger particles, with an optimal size depending on the aperture distribution of the fracture. Particles are preferentially deposited in areas with low flow velocities.

Studies often consider a limited number of parameters to verify DTs (like the permeability in e.g., Ramandi et al. 2017). However, the results from this research indicate that additional data besides permeability and mean aperture might be necessary to properly validate fracture DTs. This insight should be considered in future studies to prove that DTs of fractures properly represent the real complexity. In general, the presented workflow offers a promising potential to create reliable DTs of fractures without the need for additional calibration data. Suggestions for future studies include:

- The influence of the PVE can be diminished, and limitations in spatial resolution of a CT-setup can be exceeded by utilizing sub-resolution information with the CFMA approach.
- Future studies might focus on benchmarking the presented workflow by analysing suitable samples and comparing the results with other methods and a ground truth.
- 3-D printed fractures with pre-defined properties (e.g., a constant aperture) that allow controlling the influence of the PVE and will provide a ground truth as a reference. The presented workflow can be applied to investigate the performance of the reconstruction method in different aperture to (resampled) resolution ratios (i.e., in that way, it could be evaluated if increasing the resampled resolution increases the quality of the reconstructed fracture). Furthermore, errors from the CFMA calculation could be quantified.
- Alternatively, natural fractures with a smaller influence of the PVE as the here presented samples can be investigated.
- Results from the reconstruction workflow could be compared to other imaging techniques (like SEM) or reconstruction (based on calibration data) methods.

- In sufficiently resolved fractures (with a low influence of the PVE), the CFMA method and reconstruction workflow (without enhancing the resolution) might provide a reference to quantitatively compare different segmentation results. Furthermore, this reference can provide a suitable ground truth for training segmentation CNNs.
- An adjusted version of the code could also be applied to reconstruct fracture networks.
- Additionally, an increasing computational power might enable future studies to resample the spatial resolution by a factor greater than four and thus achieve a better representation of real aperture distributions (if the error of local aperture calculation is small enough).
- Future studies might add more complexity to the presented particle deposition models, e.g., by: (i) considering additional forces like particle charges or gravity, (ii) including different particle morphologies and interactions between particles, (iii) allowing particles to be released after being deposited (iv) considering the isolated effect of fracture properties (roughness, aperture distribution).
- Additionally, a quantitative comparison to laboratory data would help to validate the modelling results and optimize the modelling parameters. Validation data could include a quantitative analysis of size, amount (particle recovery rate), and location of deposited particles (e.g., by SEM). DRP studies might also benefit from reference samples (with a simple geometry and defined properties) that can be used to validate each step of the analysis and to optimize modelling parameters.

This study focused on the particle migration through cm-sized fractured cores, with a neglectable matrix permeability (and therefore a neglectable particle migration through the rock matrix). However, to understand particle migration in reservoir rocks and utilize them properly and sustainable (e.g., for geothermal plants or carbon capture and storage), it is crucial to study these processes in different scales, different rock types and in porous media. Additionally, insights gained from these different approaches should be combined into models where the porous rock matrix is coupled with fractures. The ongoing joint-project ResKin_Move ("Einfluss von Partikelmobilisierung infolge von Kalzitzement-Auflösung auf die Reservoireigenschaften", a follow-up project of the ResKin project) aims to complement the findings of the Resalt project (and vice versa) by focusing on the particle migration in the pore space. For example, the insights of the segmentation method comparison, presented in section 4.1, are directly applied to the samples analysed in the ResKin_Move project. The goal of this project is the investigation of the influence of fine mobilization due to dissolution processes in the pore space (e.g., Othman et al. 2018). Since results from the previous project ResKin indicated that clay minerals might be mobilised due to calcite-cement dissolution, the focus of the follow-up project is the comprehension of the permeability/porosity response due to the deposition of these clay particles. Thus, the project ResKin_Move investigates similar processes as presented in this study, but in a smaller length scale (mm-sized sampled), with different particles (Illites) and with the focus on the pore space (see Table 12 for a comparison of the analysed systems in the ReSalt and ResKin_Move project). The small size of the samples allows the usage of Micro-XCT and synchrotron-based XCT scanners, enabling high spatial resolutions (~1 – 2.5 µm) and, thus, highly detailed digital twin models. Furthermore, in-situ flow-through experiments (with clay particle injection) were conducted at a synchrotron facility to achieve high quality, time-resolved datasets. A novel particle deposition method will be applied on the digital twin models to simulate the clay particle deposition in the pore space and reproduce the resulting effect on porosity and permeability, which was observed in these experiments (the code is described in Sadeghnejad et al. 2022). In contrast to the fractured samples, the permeabilities of the porous samples are several orders of magnitude lower, indicating that particle deposition should have a greater influence on the hydraulic properties of the porous samples. Moreover, while diffusion was neglectable for the migration of the baryte particles through the fractures, the diffusion must be considered to adequately describe the transport behaviour of clay particles in the porous samples (due to smaller particle sizes, the smaller scale of the transport paths and relatively low flow rates). Therefore, the combined insights of the ReSalt and ResKin_Move projects (and follow-up investigations, e.g., considering fracture networks) allow a greater and comprehensive understanding of the fundamental processes of particle migration in geological systems. However, it is still a major challenge to combine these insights into one model, that simulates the particle transport in fractures and the surrounding rock matrix.

Table 12. Comparison of the analysed systems in the ReSalt and ResKin projects.

	Fractured samples	Porous samples
Sample size (length x diameter)	15 x 10 cm	10 x 4 mm
Spatial resolution [µm]	200	1 – 2.5
XCT Scanner	Medical XCT	Micro-/synchrotron-based XCT
Permeability range [m²]	10^{-10} – 10^{-09}	10^{-14} – 10^{-13}
Particle type	Baryte	Illite
Particle size [µm]	1 – 42	~1
Important transport mechanisms	Mostly advection dominated	Advection + Diffusion
Project	ReSalt	ResKin & the follow-up project ResKin_Move

To conclude, the results from this thesis demonstrate that DRP simulations can reproduce the complex interactions between fractures and (baryte/tracer-) particles that were observed in laboratory experiments. It is possible to use insights from these simulations to support the planning of additional experiments. This was feasible, although the representation of the fracture morphologies was limited by the spatial resolution of the XCT-scanner and various simplifications were applied for the simulations. However, extracting the fracture morphologies from the XCT-images and setting up the simulations is still a complex and laborious task, requiring sufficient computational power and expert knowledge to choose, adapt and apply the proper methods for each processing step (e.g., segmentation). In this regard, more research is required to improve existing methods and develop new methods (e.g., based on machine learning approaches) in order to overcome current limitations and make the application more efficient and user-friendly. Numerous studies demonstrated that the (usually non-destructive) DRP workflow is a great tool to analyse complex processes in real geometries. Hence, DRP is routinely used to compliment laboratory experiments and sometimes to replace additional experiments. Currently, DRP is not able to completely replace expensive and time-consuming laboratory experiments since data obtained from these is still mandatory to validate the digital models and the corresponding simulations. To reach this goal, more research is needed to e.g., (i) fully understand the complex interactions in porous/fractured media, (ii) adapt the digital models to properly consider these complexities (e.g., particle morphology, particle-particle interactions, coupling of fractures with a porous matrix), (iii) improve details in the 3-D models (e.g., fracture roughness) and (iv) increase the computational power and solver efficiency to enable the simulations of these complex models.

Appendix

```python

# Measure real fracture aperture after Huo et al. 2016. Only for approximately horizontal fractures (parallel to x-axis). Input: Tif stack
# Use a local CT mat value

import tifffile as tiff
import math

CT_air = 1013 # calculated from Avizo
R = 200 # resolution in [µm]
xy_dim = 570 # image dimensions in x and y direction

sample = "Fle_8_S2" # sample name

path = "C:\\Users\\marce\\Desktop\\Aperture_Huo_etal_2016\\" + sample + "\\Tifs_short_test\\" + sample + "_" # path to isolated grey value fracture images (masked, 0 = background & matrix that is not influenced by the fracture)
savepath = "C:\\Users\\marce\\Desktop\\Aperture_Huo_etal_2016\\" + sample + "\\Tifs_skel_local_200um_test\\" # path where results are saved
path_grey = "C:\\Users\\marce\\Desktop\\Aperture_Huo_etal_2016\\" + sample + "\\Tifs_grey_200um\\" + sample + "_" # path to grey value images (unmasked)

def aperture_measure(number): # function to measure the local apertures of one image. Input: number of the image (of the image stack) to be analysed
    if number < 10:
        slide = tiff.imread(path + "00" + str(number) + ".tif") # load current slide of isolated grey value fracture image stack
        slide_grey = tiff.imread(path_grey + "00" + str(number) + ".tif") # load current slide of grey value image stack
    elif number < 100:
        slide = tiff.imread(path + "0" + str(number) + ".tif")# load current slide of isolated grey value fracture image stack
        slide_grey = tiff.imread(path_grey + "0" + str(number) + ".tif")# load current slide of grey value image stack
    else:
        slide = tiff.imread(path + str(number) + ".tif")# load current slide of isolated grey value fracture image stack
        slide_grey = tiff.imread(path_grey  + str(number) + ".tif")# load current slide of grey value image stack

    skeleton = slide * 0 # initialize empty slide to store skeleton + aperture values inside
    for j in range(1,xy_dim): # iterate through each column of the current images (X-direction)
        Summe = 0 # keep track of the sum of CT values of fracture pixels
        start = 0 #start of fracture
        end = 0 #end of fracture
        CT_mat_sum = 0 # keep track of the local CT mat values of the matrix pixels
        matrix_count = 0 # number of matrix pixels considered for the calculation of the local CT mat value
        for i in range(1,xy_dim): #iterate through the rows in the current column of the images (Y-direction)
            if slide[i,j] > 10: # check if the current pixel belongs to the matrix/background (=0) or isolated fracture (>0)
                if start == 0: # if this is the first non-zero pixel in the column
                    start = i # save the pixel position where the isolated fracture starts in the current column
                else:
                    end = i # save the current pixel position as the end of the isolated fracture, is updated when the next pixel belongs to the fracture
        for k in range(start-4,start+1): # calculate the local CT mat value by considering the grey values of 5 background voxels above (and including) the starting pixel
            CT_mat_sum += slide_grey[k,j] # add grey value of the matrix pixel to the CT_mat_sum
            matrix_count += 1 # increase ocunter of pixels considered in the calculation of the local CT mat value
        for a in range(end,end+5): # calculate the local CT mat value by considering the grey values of 5 matrix voxels below (and including) the end pixel
            CT_mat_sum += slide_grey[a,j] # add grey value of the matrix pixel to the CT_mat_sum
            matrix_count += 1 # increase ocunter of pixels considered in the calculation of the local CT mat value
        CT_mat = CT_mat_sum/matrix_count # calculate the local CT mat value
        for voxel in range(start+1,end): # iterate through the pixels which belong to the isolated fracture
            if slide[voxel,j] < CT_mat and slide[voxel,j] > 10: # check if the pixel is influenced by the presence of the fracture (grey value < local CT mat)
               Summe += CT_mat - slide[voxel,j] # if the pixel is influenced by the fracture, save the missing attenuation (difference in grey values)

        if (end - start) > 0: # if a fracture was present in the current column
            position = start + math.ceil((end - start)/2) # determine where the fracture starts, round to upper value, to make assignment consistent
            skeleton[position,j] = (R * Summe)/(CT_mat - CT_air) # assign aperture value to fracture middle point
            if skeleton[position,j] < 75:                      # grow the skeleton according to the aperture value, grow a bit more to keep connectivity of structure
                skeleton[(position-1):(position+1),j] = skeleton[position,j]
            elif skeleton[position,j] < 125:
                skeleton[(position-1):(position+2),j] = skeleton[position,j]
            elif skeleton[position,j] < 175:
                skeleton[(position-2):(position+2),j] = skeleton[position,j]
            elif skeleton[position,j] < 225:
                skeleton[(position-2):(position+3),j] = skeleton[position,j]
            elif skeleton[position,j] < 275:
                skeleton[(position-3):(position+3),j] = skeleton[position,j]
            elif skeleton[position,j] < 325:
                skeleton[(position-3):(position+4),j] = skeleton[position,j]
            elif skeleton[position,j] < 375:
                skeleton[(position-4):(position+4),j] = skeleton[position,j]
            elif skeleton[position,j] >= 375:
                skeleton[(position-4):(position+5),j] = skeleton[position,j]

        localsavepath = savepath + sample + "_" + "local_"  + str(number)+ ".tif"  # create path to save the result
        try:
            tiff.imsave(localsavepath,skeleton) # save the result
        except:
            print("error") # otherwise give an error

    print(str(number)) # print the number of the analysed image


for k in range(0,720): #execute the aperture_measure function for an image stack, starting from the first image (0) to the last image (here 719)
    aperture_measure(k)
```

Figure 43. Python script to calculate local apparent fracture apertures with a local CT_{mat} value.

```

# Measure real fracture aperture after Huo et al. 2016. Only for horizontal fractures (parallel to x-axis). Input: Tif stack
# Use a global CT mat value
import tifffile as tiff
import math

CT_air = 1012 # calculated from Avizo
CT_mat = 3685 # calculated from Avizo, Fle_7_S1:3666 , Fle_8_S1:3691, Fle_8_S2: 3672, Fle_9_S1: 3686, Fle_9_S2: 3685
              # Rem 1 S2: 3586, Rem 1 S1: 3585, Rem 1 S3: 3576
R = 200 # resolution in [µm]
xy_dim = 570 # image dimensions in x and y direction
sample = "Fle_8_S2" # sample name
path = "C:\\Users\\marce\\Desktop\\Aperture_Huo_etal_2016\\" + sample + "\\Tifs_short_2\\" + sample + "_" # path to grey value images (unmasked)
savepath = "C:\\Users\\marce\\Desktop\\Aperture_Huo_etal_2016\\" + sample + "\\Tifs_skel_big_200um_2\\" # path where results are saved


def aperture_measure(number): # function to measure the local apertures of one image. Input: number of the image (of the image stack) to be analysed
    if number < 10:
        slide = tiff.imread(path + "00" + str(number) + ".tif") # load current slide of grey value fracture image stack
    elif number < 100:
        slide = tiff.imread(path + "0" + str(number) + ".tif") # load current slide of grey value fracture image stack
    else:
        slide = tiff.imread(path + str(number) + ".tif") # load current slide of grey value fracture image stack

    skeleton = slide * 0 # initialize empty slide to store skeleton + aperture values inside
    for j in range(1,xy_dim): #iterate through the columns of the images (X-direction)
        Summe = 0 # keep track of the sum of CT values of fracture pixels
        start = 0 #start of fracture
        end = 0 #end of fracture
        for i in range(1,xy_dim): #iterate through the rows in the current column of the images (Y-direction)
            if slide[i,j] > 10 and slide[i,j] < CT_mat: # check if the current pixel belongs to the background (=0) or fracture (>0 and < global CT_mat value)
                if Summe == 0: # if this is the first pixel in the current column, belonging to the fracture
                    start = i # save the pixel position where the isolated fracture starts in the current column
                else:
                    end = i # save the current pixel position as the end of the isolated fracture, is updated when the next pixel belongs to the fracture
                Summe += CT_mat - slide[i,j] # if the pixel is influenced by the fracture, save the missing attenuation (difference in grey values)
        if (end - start) > 0: # if a fracture was present in the current column
            position = start + math.ceil((end - start)/2) # determine where fracture starts, round to upper value, to make assignment consistent
            skeleton[position,j] = (R * Summe)/(CT_mat - CT_air) # assign aperture value to fracture middle point
            if skeleton[position,j] < 75:                     # grow the skeleton according to the aperture value, grow a bit more to keep connectivity of structure
                skeleton[(position-1):(position+1),j] = skeleton[position,j]
            elif skeleton[position,j] < 125:
                skeleton[(position-1):(position+2),j] = skeleton[position,j]
            elif skeleton[position,j] < 175:
                skeleton[(position-2):(position+2),j] = skeleton[position,j]
            elif skeleton[position,j] < 225:
                skeleton[(position-2):(position+3),j] = skeleton[position,j]
            elif skeleton[position,j] < 275:
                skeleton[(position-3):(position+3),j] = skeleton[position,j]
            elif skeleton[position,j] < 325:
                skeleton[(position-3):(position+4),j] = skeleton[position,j]
            elif skeleton[position,j] < 375:
                skeleton[(position-4):(position+4),j] = skeleton[position,j]
            elif skeleton[position,j] >= 375:
                skeleton[(position-4):(position+5),j] = skeleton[position,j]

        localsavepath = savepath + sample + "_" + str(number)+ ".tif"  # create path to save the result
        try:
            tiff.imsave(localsavepath,skeleton) # save the result
        except:
            print("error") # otherwise throw an error

    print(str(number)) # print the number of the analysed image


for k in range(0,720): #execute the aperture_measure function for an image stack, starting from the first image (0) to the last image (here 719)
    aperture_measure(k)
```

Figure 44. Python script to calculate local apparent fracture apertures with a global CT_{mat} value.

```
clear all
close all
sample = 'Fle_9_S1_' % sample name
cd /DATA/BIGRAID/RESALT/skeletonized_fracs/Fle_9_S1_Huo   %change directory to data folder


i=-1; % keep track of current image
location = 'Tifs_grey_50um_seg/*tif' %  folder in which the binary images exists; 0 = matrix, 1 = fracture, 50µm resolution
ds = imageDatastore(location)         %  Creates a datastore for all images in the folder
while hasdata(ds)
    img = read(ds) ;             % read image from datastore
    i=i+1 %count number of iterations to name skeletonized files later
    img= img >0; %convert image_type to logical
    B = bwskel(img,'MinBranchLength',25); %skeletonize 2D image
    string = num2str(i); %prepare filename
    if i < 10
        path = ['Skeleton/' sample '50um_skel_000' string '.tif']; %path to store skeletonized tif, numbered by iteration i
    elseif i < 100
        path = ['Skeleton/' sample '50um_skel_00' string '.tif']; %path to store skeletonized tif, numbered by iteration i
    elseif i < 1000
        path = ['Skeleton/' sample '50um_skel_0' string '.tif']; %path to store skeletonized tif, numbered by iteration i
    elseif i >= 1000
        path = ['Skeleton/' sample '50um_skel_' string '.tif']; %activate when you have more than 1000 images (at 50um resolution) and update zeros in the above paths
    end

    imwrite(B,path); %save image
end
```

Figure 45. Matlab script for calculating the fracture skeleton.

```
% assign aperture value to skeleton and grow the skeleton

clear all
close all

cd /DATA/BIGRAID/RESALT/skeletonized_fracs/Fle_8_S1_Huo   %change directory to data folder
i=-1; % counter for keeping track of the current image number
location = 'Skeleton/*tif'  %  folder in which the skeleton images exists
location_grey = 'Tifs_grey_50um/*tif'  %  folder in which the grey value aperture images exists
growing_limit = 75; %determines the growing of the skeleton

ds_skel = imageDatastore(location)         %  Creates a datastore for all skeleton images in the folder
ds_grey = imageDatastore(location_grey)         %  Creates a datastore for all grey value images in the folder


while hasdata(ds_skel) %import the grey value images and the skeleton images
    img_skel = read(ds_skel); %read the next skeleton image
    img_grey = read(ds_grey); %read the next grey value image
    i = i+1 %keep track of current image number
    img_grow = img_grey*0; %create a new image for the grown skeleton, with the same dimensions as the grey value image
    positions = find(img_skel >0); %find the position of the pixels with a value > 0 in the skeleton images (i.e., pixels belonging to the skeleton)



%%%%%%%%% START skeleton growing procedure alternative 1: with if statements. When uncommented, the growing procedure alternative 2 must be commented
%%%%%%%%% This procedure was applied in the thesis

    % for j = drange(1:length(positions)) %grow the skeleton at each position, according to the stored local aperture value
        % if img_grey(positions(j)) < growing_limit
            % img_grow(positions(j)) = img_grey(positions(j)); %save the local fracture positions in the new image stack
        % elseif img_grey(positions(j)) < (growing_limit + 1*50)
            % img_grow(positions(j)-1:positions(j)) = img_grey(positions(j));
        % elseif img_grey(positions(j)) < (growing_limit + 2*50)
            % img_grow(positions(j)-1:positions(j)+1) = img_grey(positions(j));
        % elseif img_grey(positions(j)) < (growing_limit + 3*50)
            % img_grow(positions(j)-2:positions(j)+1) = img_grey(positions(j));
        % elseif img_grey(positions(j)) < (growing_limit + 4*50)
            % img_grow(positions(j)-2:positions(j)+2) = img_grey(positions(j));
        % elseif img_grey(positions(j)) < (growing_limit + 5*50)
             % img_grow(positions(j)-3:positions(j)+2) = img_grey(positions(j));
        % elseif img_grey(positions(j)) < (growing_limit + 6*50)
            % img_grow(positions(j)-3:positions(j)+3) = img_grey(positions(j));
        % elseif img_grey(positions(j)) < (growing_limit + 7*50)
            % img_grow(positions(j)-4:positions(j)+3) = img_grey(positions(j));
        % elseif img_grey(positions(j)) < (growing_limit + 8*50)
            % img_grow(positions(j)-4:positions(j)+4) = img_grey(positions(j));
        % elseif img_grey(positions(j)) < (growing_limit + 9*50)
            % img_grow(positions(j)-5:positions(j)+4) = img_grey(positions(j));
        % elseif img_grey(positions(j)) < (growing_limit + 10*50)
            % img_grow(positions(j)-5:positions(j)+5) = img_grey(positions(j));
        % elseif img_grey(positions(j)) < (growing_limit + 11*50)
             % img_grow(positions(j)-6:positions(j)+5) = img_grey(positions(j));
        % elseif img_grey(positions(j)) < (growing_limit + 12*50)
            % img_grow(positions(j)-6:positions(j)+6) = img_grey(positions(j));
        % elseif img_grey(positions(j)) < (growing_limit + 13*50)
            % img_grow(positions(j)-7:positions(j)+6) = img_grey(positions(j));
        % elseif img_grey(positions(j)) < (growing_limit + 14*50)
```

```
            % img_grow(positions(j)-7:positions(j)+7) = img_grey(positions(j));
        % elseif img_grey(positions(j)) < (growing_limit + 15*50)
            % img_grow(positions(j)-8:positions(j)+7) = img_grey(positions(j));
        % elseif img_grey(positions(j)) < (growing_limit + 16*50)
            % img_grow(positions(j)-8:positions(j)+8) = img_grey(positions(j));
        % elseif img_grey(positions(j)) < (growing_limit + 17*50)
            % img_grow(positions(j)-9:positions(j)+8) = img_grey(positions(j));
        % elseif img_grey(positions(j)) < (growing_limit + 18*50)
            % img_grow(positions(j)-9:positions(j)+9) = img_grey(positions(j));
        % elseif img_grey(positions(j)) < (growing_limit + 19*50)
             % img_grow(positions(j)-10:positions(j)+9) = img_grey(positions(j));
        % elseif img_grey(positions(j)) < (growing_limit + 20*50)
            % img_grow(positions(j)-10:positions(j)+10) = img_grey(positions(j));
        % elseif img_grey(positions(j)) < (growing_limit + 21*50)
            % img_grow(positions(j)-11:positions(j)+10) = img_grey(positions(j));
        % elseif img_grey(positions(j)) < (growing_limit + 22*50)
            % img_grow(positions(j)-11:positions(j)+11) = img_grey(positions(j));
        % elseif img_grey(positions(j)) < (growing_limit + 23*50)
            % img_grow(positions(j)-12:positions(j)+11) = img_grey(positions(j));
        % elseif img_grey(positions(j)) < (growing_limit + 24*50)
            % img_grow(positions(j)-12:positions(j)+12) = img_grey(positions(j));
        % elseif img_grey(positions(j)) < (growing_limit + 25*50)
             % img_grow(positions(j)-13:positions(j)+12) = img_grey(positions(j));
        % elseif img_grey(positions(j)) < (growing_limit + 26*50)
            % img_grow(positions(j)-13:positions(j)+13) = img_grey(positions(j));
        % elseif img_grey(positions(j)) < (growing_limit + 27*50)
            % img_grow(positions(j)-14:positions(j)+13) = img_grey(positions(j));
        % elseif img_grey(positions(j)) < (growing_limit + 28*50)
            % img_grow(positions(j)-14:positions(j)+14) = img_grey(positions(j));
        % elseif img_grey(positions(j)) < (growing_limit + 29*50)
            % img_grow(positions(j)-15:positions(j)+14) = img_grey(positions(j));
        % elseif img_grey(positions(j)) < (growing_limit + 30*50)
            % img_grow(positions(j)-15:positions(j)+15) = img_grey(positions(j));
        % elseif img_grey(positions(j)) < (growing_limit + 31*50)
            % img_grow(positions(j)-16:positions(j)+15) = img_grey(positions(j));
        % elseif img_grey(positions(j)) < (growing_limit + 32*50)
            % img_grow(positions(j)-16:positions(j)+16) = img_grey(positions(j));
        % elseif img_grey(positions(j)) < (growing_limit + 33*50)
            % img_grow(positions(j)-17:positions(j)+16) = img_grey(positions(j));
        % elseif img_grey(positions(j)) < (growing_limit + 34*50)
            % img_grow(positions(j)-17:positions(j)+17) = img_grey(positions(j));
        % elseif img_grey(positions(j)) < (growing_limit + 35*50)
             % img_grow(positions(j)-18:positions(j)+17) = img_grey(positions(j));
        % elseif img_grey(positions(j)) < (growing_limit + 36*50)
            % img_grow(positions(j)-18:positions(j)+18) = img_grey(positions(j));
        % elseif img_grey(positions(j)) < (growing_limit + 37*50)
            % img_grow(positions(j)-19:positions(j)+18) = img_grey(positions(j));
        % elseif img_grey(positions(j)) < (growing_limit + 38*50)
            % img_grow(positions(j)-19:positions(j)+19) = img_grey(positions(j));
        % elseif img_grey(positions(j)) < (growing_limit + 39*50)
            % img_grow(positions(j)-20:positions(j)+19) = img_grey(positions(j));
        % elseif img_grey(positions(j)) < (growing_limit + 40*50)
            % img_grow(positions(j)-20:positions(j)+20) = img_grey(positions(j));
        % elseif img_grey(positions(j)) >= (growing_limit + 40*50)
            % img_grow(positions(j)-21:positions(j)+20) = img_grey(positions(j));
        % end
    % end

%%%%%%%%% END skeleton growing procedure alternative 1: with if statements. When uncommented, the growing procedure alternative 2 must be commented
```

```


%%%%%%%%% START skeleton growing procedure alternative 2: with while statement. When uncommented, the growing procedure 1 must be commented
%%%%%%%%% This procedure is more general and efficient than the growing procedure alternative 1

    for j = drange(1:length(positions)) % grow the skeleton at each position, according to the stored local aperture value
        growing_limit_local = growing_limit; % new variable which increases by 50 µm during each iteration of the while loop
        k = 0; % keep track of the number of pixels, that will be added locally to the skeleton
        while img_grey(positions(j)) >= growing_limit_local % compare the local aperture in the grey value images with the current growing_limit_local
            growing_limit_local = growing_limit_local + 50; % increase growing_limit by 50 µm (size of a pixel in the resampled resolution)
            k = k +1; % increase the number of pixels, that will be added locally to the skeleton
        end
        if mod(k,2) == 0 % check if an equal number of pixels will be added locally to the skeleon
            img_grow(positions(j)-(k/2):positions(j)+(k/2)) = img_grey(positions(j)); % add k/2 pixels below and above the local skeleton pixel
        elseif mod(k,2) == 1 % check if an unequal number of pixels will be added locally to the skeleon
            k = k - 1; % decrease the number of pixels to be added by 1
            img_grow(positions(j)-(1+k/2):positions(j)+(k/2)) = img_grey(positions(j)); % add k/2 pixels above the local skeleton pixel, and k/2 + 1 pixels below the local skeleton pixel
        end
    end

%%%%%%%%% END  skeleton growing procedure alternative 2: with while statement. When uncommented, the growing procedure 1 must be commented



    string = num2str(i); %prepare filename
    if i < 10
        path = ['Skeleton grow/Fle 8 S1 50um skelgr 000' string '.tif']; %path to store skeletonized tif, numbered by iteration i
    elseif i < 100
        path = ['Skeleton grow/Fle 8 S1 50um skelgr 00' string '.tif']; %path to store skeletonized tif, numbered by iteration i
    elseif i < 1000
        path = ['Skeleton grow/Fle 8 S1 50um skelgr 0' string '.tif']; %path to store skeletonized tif, numbered by iteration i
    elseif i >= 1000
        path = ['Skeleton grow/Fle 8 S1 50um skelgr ' string '.tif']; %path to store skeletonized tif, numbered by iteration i
    end
    imwrite(img grow,path); %save image
end
```

Figure 46. Matlab script to grow a fracture skeleton according to the local apparent aperture.

```python
### Dr. Francesco Cappuccio adaped his original code to measure the aperture of segmented fractures.
### His original code is published in Cappuccio F, Toy VG, Mills S, Adam L (2020) https://doi.org/10.3389/feart.2020.529263
### To apply the code you have to run the following scripts: Step0, Step1 and finally Step2
### EDIT path_0 and path_f, the images have to be stored in a folder called "images" and to save the results,
### you have to create a folder called "results"
import os
import numpy as np
from glob import glob
from skimage import io
from scipy import ndimage as ndi
#import pylab as plt
def s():
    import sys
    sys.exit()
#################################################################################################################################
"""Convert binary images to grayscale ones using distance map"""

path_0="C:\\Users\\marce\\Desktop\\apertures_laserscans_francesco_script\\Fle_8_S1\\images"
path_f="C:\\Users\\marce\\Desktop\\apertures_laserscans_francesco_script\\Fle_8_S1\\results\\"
fldrs = os.listdir(path_0)
for fldr in fldrs:
    print("Analyzing folder {}...".format(fldr))
    name =fldr
    os.chdir(path_0+"\\"+name)
    imgs = list(glob("*.tif"))
    img3d=[]
    for i in imgs:
        img3d.append(io.imread(i))
    img3d = np.array(img3d, dtype="uint8", copy=False)
    img3d[img3d>0]=1
    #plt.figure()
    #ax1 =plt.subplot(121)
    #plt.imshow(img3d[0])
    #plt.subplot(122, sharex = ax1, sharey = ax1)
    img2=[]
    for i in img3d:
        img2.append(ndi.distance_transform_edt(i)*100)
    os.chdir(path_f)
    os.makedirs(name)
    os.chdir(path_f+name)
    os.makedirs ("msk")
    os.chdir(path_f+name+'\\'+"msk")
    io.imsave("msk.tif",img3d)
    del img3d
    print ("Distance-map created")
    img2=np.array(img2, copy=False)
    mx = np.amax(img2)
    #img2[img2==0]=np.nan
    from skimage import util
    img2=util.invert(img2)
    img2=img2+mx
    #img2=np.nan_to_num(img2, copy=False).astype(int)
    vals = np.unique(img2)
    for n,i in enumerate(vals):
    #    if i>0:
        img2[img2==i]=n
    #    else:
    #        img2[img2==i]=len(vals)
    img2 = img2.astype("uint8")
    print ("Grayscale image created")
    #img2 = exposure.rescale_intensity(img2, out_range='uint8')
    #plt.imshow(img2[0])
    os.chdir(path_f+name)
    os.makedirs ("img")
```

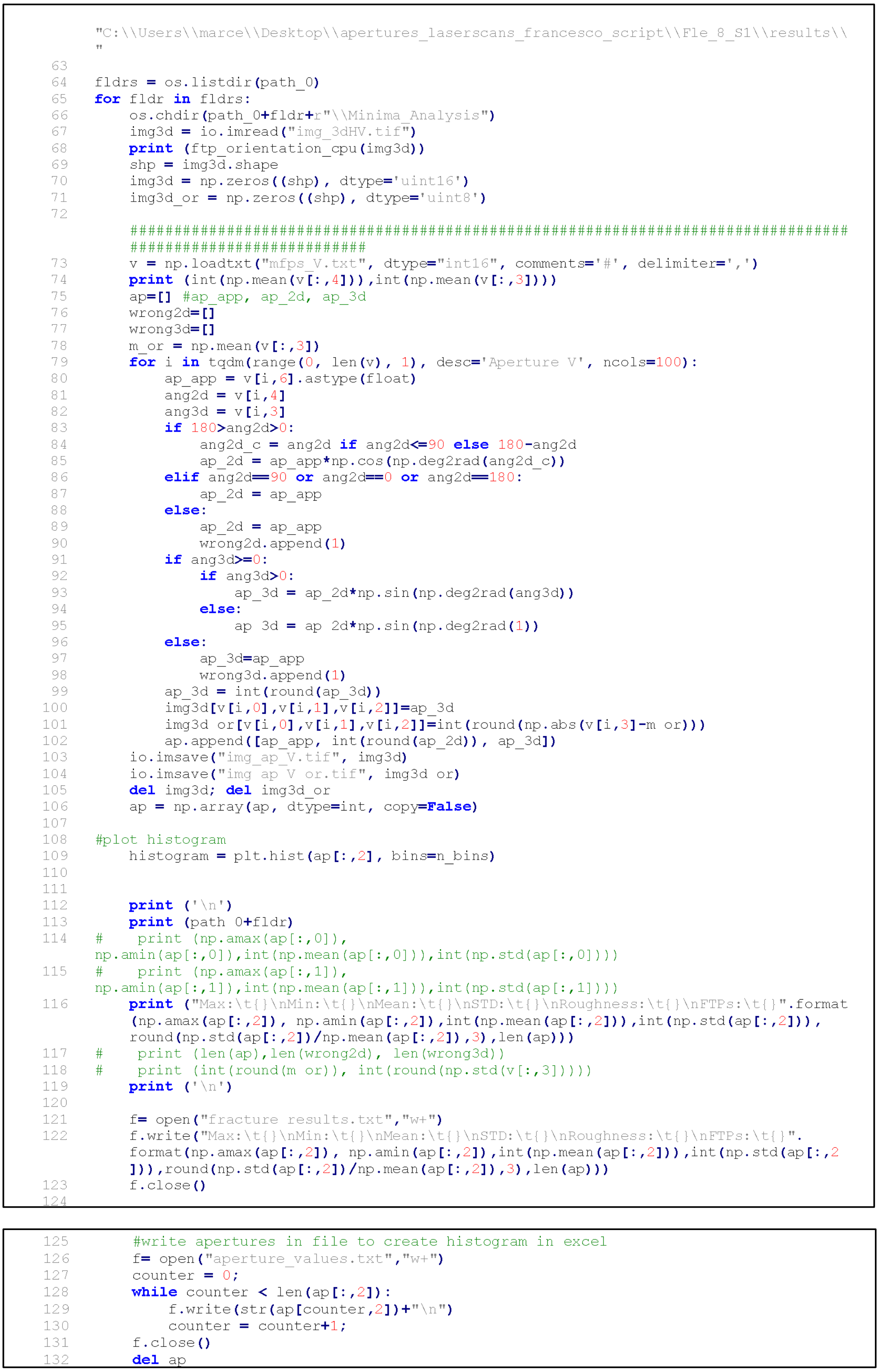
```python
    os.chdir(path_f+name+'\\'+"img")
    io.imsave("img.tif",img2)
    print("Imaged saved in {}\n\n".format(path_f))
```

Figure 47. Python script to measure the apertures of segmented fractures (Step0).
Written by Dr. Francesco Cappuccio, the original code is described in Cappuccio et al. 2020.

```python
### Dr. Francesco Cappuccio adaped his original code to measure the aperture of segmented fractures.
### His original code is published in Cappuccio F, Toy VG, Mills S, Adam L (2020) https://doi.org/10.3389/feart.2020.529263
### To apply the code you have to run the following scripts: Step0, Step1 and finally Step2
### EDIT the resolution (last line) to your image resolution
### path_0, to the results folder used in the previous step (Step0)
import timeit
import os
from glob import glob
import numpy as np
import sys
from skimage import io
import psutil
from joblib import Parallel, delayed
from tqdm import tqdm
import scipy.signal
import math
def s():
    sys.exit()
import pylab as plt
#
def analyze_image_stack (path, vox_dim, matrix=None, air=None, res=None, z_corr=None):
    """ The program runs on a stack of .tif images and return measurements on every Fracture Trace Point (FTP) found.
    vox_dim: size of the 3D local crop for orientation analysis
    matrix: mean matrix CT-value
    air: mean air CT-value
    res: FWHM value will only be measured for structures with an Edge Response (ER)>res and if the FTP is below the FWHM baseline value [((mtrx-air)/2)+air]
    z_corr: for cubic voxel is equal to 1 (put None in that case), for non cubic voxel (with x=y) is z/x
    It returns:
    - 3D .tif image of the FTPs
    - 3 .txt files reporting measurements of each FTP (position, CT-value, orientation of the best fitting plane, Edge Response (ER), FWHM, PH, MA)
    """
    start = timeit.default_timer() #Start timing the function
    np.set_printoptions(threshold=sys.maxsize, linewidth=200)
    path0 = os.getcwd ()
    path_img = ("".join ([path, "\\img"])) #Path to the folder where the images are stored
    path_mask = ("".join ([path, "\\msk"])) #Path to the folder where the binary images of the fractures are stored
    path_store = ("".join ([path, "\\Minima_Analysis"])) #Path to the folder where the program store the data
    os.chdir (path_mask) #Change path to the folder where the mask_images are located
    mask_stack = sorted (glob("*.tif")) #List of the names of the TIF files in the folder (for the mask)
    if len(mask_stack)==0:
        raise ValueError("No TIF files found in the folder msk")
    os.chdir (path_img) #Change path to the folder where the images are located
    img_stack = sorted (glob("*.tif")) #List of the names of the TIFF files in the folder (for the original image)
    for i in range (0, len(mask_stack), 1): #List of the names of the TIFF files for the mask (including the path)
        mask_stack[i] = ("".join ([path_mask, "\\", mask_stack[i]]))
    if len(mask_stack)==0:
        raise ValueError("No TIF files found in the folder img")
    os.chdir(path)
    if os.path.exists (path_store): #Check if the folder Minima_Analysis exists and has file inside (they will be removed).
        if len(os.listdir(path_store))>0:
            for f in os.listdir(path_store):
                os.remove(os.path.join(path_store, f))
    else:
        os.makedirs ("Minima_Analysis") #Create a folder to store the data inside the original image folder
    img3D = [] #An empty list for appending the images while opening slices
    img_mask3D = [] #An empty list for appending the mask_images while opening slices
    #Check the module to use for opening the images
    os.chdir (path_img)
    #Tries different modules of io.imread to open the images with correct shape
    modules = ['imageio','pil','tifffile','matplotlib']
    module_i=[]
    module_m=[]
    for n, i in enumerate(modules):
        if np.array(io.imread(img_stack[0], plugin=i)).ndim==3 and len(module_i)==0:
            module_i = modules[n]
        if np.array(io.imread(mask_stack[0], plugin=i)).ndim==3 and len(module_m)==0:
            module_m = modules[n]
    if not module_i or not module_m:
        raise ValueError("Cannot open mask properly with skimage.io")
    #Open the Images
    img3D = io.imread(img_stack[0])
    img_mask3D = io.imread(mask_stack[0])
    del img_stack; del mask_stack #Delete useless variables to empty the RAM
    img3D = np.array (img3D, copy=False)#[-40:] #Convert list to a numpy array
    img_mask3D = np.array (img_mask3D, copy=False, dtype="uint8")#[-40:] #Convert list to a numpy array
    print ("Image shape is\tz: {}\ty: {}\tx: {}\n".format(img3D.shape[0], img3D.shape[1], img3D.shape[2]))

#    img3D = np.rot90(img3D, k=3, axes=(1, 2))
#    img_mask3D = np.rot90(img_mask3D, k=3, axes=(1, 2))

#    img3D[:,0,:] = 0; img3D[:,-1,:] = 0; img3D[:,:,0] = 0; img3D[:,:,-1] = 0 #The grayscale image must have a 0 border
    img3D_rot =  np.rot90 (img3D, k=1, axes=(2,1)) #Rotate the 3D image for V analysis, step 1
    img3D_rot = img3D_rot[:,:,::-1] #Rotate the 3D image for V analysis, step 2
    # Analysis minima ########################################################################################################
    arr_minH3D, dataH = analyze_profile (img3D, img_mask3D, res, md='H')
    img_mask3D_rot = img_mask3D.copy()
#    img_mask3D_rot[np.where(arr_minH3D>0)] = 0 #Avoid re-measuring same FTPs
    img_mask3D_rot =  np.rot90 (img_mask3D_rot, k=1, axes=(2,1)) #Rotate the 3D mask_image for vertical analysis, step 1
    img_mask3D_rot = img_mask3D_rot[:,:,::-1] #Rotate the 3D mask_image for vertical analysis, step 2
    arr_minV3D_rot, dataV = analyze_profile (img3D_rot, img_mask3D_rot, res, md='V')
    del img3D; del img_mask3D; del img3D_rot; del img_mask3D_rot #Delete useless variables to empty the RAM
    # H, V, and HV minima images ############################################################################################
    os.chdir (path_store) #Change directory in order to store all data
    arr_minH3D = np.array (arr_minH3D, dtype="uint8", copy=False) #Convert the list of minima images to a 3D array
    arr_minV3D_rot = np.array (arr_minV3D_rot, dtype="uint8", copy=False) #Convert the list of minima_images to a 3D array
    arr_minV3D_rot = arr_minV3D_rot[:,:,::-1] #Rotate the 3D image to the original position, step 1
    arr_minV3D_rot = np.rot90 (arr_minV3D_rot, k=1, axes=(-2,-1)) #Rotate the 3D image to the original position, step 2
    arr_minHV3D = np.add (arr_minH3D, arr_minV3D_rot, dtype="uint8") #Add the two 3D images
    del arr_minV3D_rot; del arr_minH3D #Delete useless variables to empty the RAM
    # Pad HV_min and HV_rec with zeroes to avoid problem when cropping ######################################################
    dataH = np.array(dataH, copy=False)#.astype (int)
    dataV = np.array(dataV, copy=False)#.astype (int)
    eg_eg = vox_dim//2
    dataH[:,:3]=dataH[:,:3]+eg_eg;
    dataH = list(dataH)
    dataV[:,:3]=dataV[:,:3]+eg_eg;
    dataV = list(dataV)
    arr_minHV3D = np.pad(arr_minHV3D, ((eg_eg,eg_eg), (eg_eg,eg_eg), (eg_eg, eg_eg)), mode='constant', constant_values=0)
    # Orientation Analysis ###################################################################################################
    elapsed_1 = timeit.default_timer(); print ("Minima analysis took {}s".format(round(timeit.default_timer() - start), 2))
    print ("\nCalculating orientations on {} points...".format(len(dataH)+len(dataV)))

    if psutil.virtual_memory()[2]>85: raise ValueError("Low memory. {}% used.".format(psutil.virtual_memory()[2])) #Stops script if RAM is full
    half_dim = vox_dim//2
    rs = Parallel(n_jobs=-1)(delayed(ftp_orientation_cpu)(arr_minHV3D, dataH[n], vox_dim, half_dim, H=True, core=True, z_c=z_corr) for n in tqdm(range(len(dataH)), desc='Orientation analysis', ncols=100)) #CPU1
    dataH = Parallel(n_jobs=-1)(delayed(np.insert)(dataH[n], 3, rs[n]) for n in tqdm(range(len(dataH)), desc='Inserting orientation data', ncols=100)); del rs #CPU2
    elapsed_2 = timeit.default_timer(); print ("{}\tH points are done....it took {}\n".format(len(dataH), round(elapsed_2 - elapsed_1), 2)) #Stops script if RAM is full
    if psutil.virtual_memory()[2]>85: raise ValueError("Low memory. {}% used.".format(psutil.virtual_memory()[2]))
    rs = Parallel(n_jobs=-1)(delayed(ftp_orientation_cpu)(arr_minHV3D, dataV[n], vox_dim, half_dim, H=False, core=True, z_c=z_corr) for n in tqdm(range(len(dataV)), desc='Orientation analysis', ncols=100)) #CPU1
    dataV = Parallel(n_jobs=-1)(delayed(np.insert)(dataV[n][3:], 0, ([dataV[n][0], dataV[n][2], dataV[n][1], rs[n][0], rs[n][1]])) for n in tqdm(range(len(dataV)), desc='Inserting orientation data', ncols=100)); del rs  #CPU2
    elapsed_3 = timeit.default_timer(); print ("{}\tV points are done...it took {}s\n".format(len(dataV), round(elapsed_3 - elapsed_2), 2))

    ##########################################################################################################################################
    arr_minHV3D = arr_minHV3D[eg_eg:-eg_eg,eg_eg:-eg_eg,eg_eg:-eg_eg] #Crop the image to remove the zero padded before and restore shape
    io.imsave ("img_3dHV.tif", arr_minHV3D); #Save the 3d minima image as TIF
    del arr_minHV3D; #del arr_HV3D_rec #Delete useless variables to empty the RAM
    print ("\nSaving data...")
    data_format = " z,   y,   x,  da,  dd, val, ER,FWHM,  PH,  MA"
    dataH = np.array (dataH, copy=False)#.astype (int);
    dataH[:,:3]=dataH[:,:3]-eg_eg #Return correct z,y,x values after np.pad
    np.savetxt ("mfps_H.txt", dataH, fmt='%04d', delimiter=',', header=data_format)
    dataV = np.array (sorted (dataV, key=lambda x: (x[0], x[1], x[2])), copy=False)#.astype (int)
    dataV[:,:3]=dataV[:,:3]-eg_eg #Return correct z,y,x values after np.pad
    np.savetxt ("mfps_V.txt", dataV, fmt='%04d', delimiter=',', header=data_format)
    dataHV = np.concatenate ((dataH[:,:5], dataV[:,:5]), axis=0) #Concatenate array (only orientations)
    dataHV.view("i,i,i,i,i").sort(order=['f0','f1','f2'], axis=0)
    np.savetxt ("mfps_HV.txt", dataHV, fmt='%04d', delimiter=',', header=data_format[:22])
    print ("\nData H shape:\t{}\nData V shape:\t{}\nData H&V shape:\t{}\n".format(dataH.shape, dataV.shape, dataHV.shape))
    print ("\nThe function took {} seconds.\nEND".format(round(timeit.default_timer() - start),2))
    del dataH; del dataV; del dataHV
    os.chdir (path0)
#
def analyze_profile (arr3d, msk, rs, md=None):
    """Analyze a 1D profile and return:
    - a binary array with ones at valleys positions;
    - an array that contains for each valley these measurements: valley_index, value_valley, ER, FWHM, PH, MA"""
    arr_min3D = np.zeros((arr3d.shape)) #3D array of minima from H analysis
    matrix=np.amax(arr3d)
    data = []
    for n in tqdm(range(0, len(arr3d), 1), desc='Traverse Analysis '+ md, ncols=100):
        if psutil.virtual_memory()[2]>90: #If the RAM is full over the 85%, the program will stop
            raise ValueError("Low memory. {}% used.".format(psutil.virtual_memory()[2]))
        for m in range(0, len(arr3d[n])-1, 1):
            if np.count_nonzero(msk[n, m, :])>0:
                arr = arr3d[n, m, :].copy()
                msk_indx = np.nonzero (msk[n, m, :]) #Return the indices to keep (of value 1)
                arr_v = -1*arr #In order to get valleys (recorded as peaks) when applying scipy.signal.find_peaks
                height_min = np.amin(arr_v[msk_indx]) #Minimum value in the thresholded image
                height_max = (np.amax(arr_v[msk_indx]) if np.amax(arr_v[msk_indx])<0 else -1) #Maximum value in the inverted image
                valleys, _ = scipy.signal.find_peaks (arr_v, height=(height_min, height_max)) #Return peaks indexes of the arr_v (valleys of arr)
                valleys = np.intersect1d (valleys, msk_indx, assume_unique=True) #Remove indexes not present in the mask
                del arr_v; del msk_indx #Delete useless variables to empty the RAM
                true_valleys = [] #Valleys that miss an edge (because close to the background which is equal to 0)
                res = [] #average of the 10-90% of the x-distance between the valley-edge
                PH = [] #PH measured from the highest edge
                FWHM = [] #FWHM at half of PH
                MA = [] #MA (sum of the y-distances) from the highest edge to arr[n]
                if valleys.size > 0: #Check if the arrays are not empty
                    for i in valleys:
                        peaks = np.where(arr==matrix)[0]
                        peaks_l = peaks[peaks<i]
                        peaks_r = peaks[peaks>i]
                        if len(peaks_l)>0 and len(peaks_r)>0:
                            peaks_l_abs = np.abs(peaks_l-i)
                            peaks_l = peaks_l[np.where(peaks_l_abs==np.amin(peaks_l_abs))][0]
                            peaks_r_abs = np.abs(peaks_r-i)
                            peaks_r = peaks_r[np.where(peaks_r_abs==np.amin(peaks_r_abs))][0]
                            ap = peaks_r-peaks_l-1
                            true_valleys.append(i)
                            res.append(ap*rs)
                            PH.append(ap)
                            FWHM.append(ap)
                            MA.append(ap)
                            value = np.take (arr3d[n, m, :], true_valleys) #Return the value of each valley
                            arr_min3D[n, m, true_valleys] = 255 #Fill the array of 255 in the valley index
                            z = [n]*len(true_valleys)
                            y = [m]*len(true_valleys)
                            data.append (zip (z, y, true_valleys, value, res, FWHM, PH, MA)) #Put the data together in a numpy array
    flatten = [k for i in data for k in i]
    flatten = np.array(flatten); print ('')
    return arr_min3D, flatten
#
def closest_indx (x1, x2):
    """It returns from the indexes in x1, the closest to x2"""
    np.seterr(all='ignore')
    abs_=[]
    for i in x1:
        abs_.append(abs(i-x2))
    indx = abs_.index(min(abs_))
    return indx
#
def ftp_orientation_cpu(img, dt, d, h, H=True, core=True, z_c=None):
    """It measure the orientation of the plane that best fits the FTPs"""
    np.seterr(invalid='ignore');
    Z=dt[0]
    Y = dt[1] if H is True else dt[2]
    X = dt[2] if H is True else dt[1]
    arr = img[int(Z),int(Y)-h:int(Y)+(d-h), int(X)-h:int(X)+(d-h)].copy()
    orient2d=[]
    if np.count_nonzero(arr)<2: #Check if there are less than 3 points
        orient2d.append(-1.)
        pass
    else:
        y, x = np.where(arr>0)
        north = [1,0]
        x = x - np.mean(x)
        y = y - np.mean(y); y = y[::-1]
        _, _, vv = np.linalg.svd(np.concatenate((x[:, np.newaxis], y[:, np.newaxis]),
        axis=1))
        v = vv[0]
        v = v*(-1.0 if v[1]>=0.0 else 1.0)
        v1_u = north/np.linalg.norm(north)
        v2_u = v/np.linalg.norm(v)
        ang = round(np.degrees(np.arccos(np.clip(np.dot(v1_u, v2_u), -1.0, 1.0))),0)
        if v[0]>0 and ang>90.:
            ang=360-ang
        elif v[0]<0 and ang>180:
            ang=360-ang
        orient2d.append(ang)
    arr = img[int(Z)-h:int(Z)+(d-h), int(Y)-h:int(Y)+(d-h), int(X)-h:int(X)+(d-h)].copy()
    arr = np.rot90(arr, axes=(0,1))
    orient3d=[]
    if np.count_nonzero(arr)<3: #Check if there are less than 3 points
        orient3d.append([-1.,-1.])
        pass
    else:
        indx = np.nonzero(arr) #Indexes of the non-zero points
        z = indx[0]
        y = indx[1]
        x = indx[2]
        x = x - np.mean(x)
        y = y - np.mean(y);# y = y[::-1]
        z = z - np.mean (z); #To center the cloud of points
        evals, evecs = np.linalg.eig(np.cov([x, y, z])) #Calculate eigenvalues and eigenvectors of the covariance matrix
        sort_indices = np.argsort(evals)[::-1]
        x_v3, y_v3, z_v3 = evecs[:, sort_indices[2]]
        north = np.asarray ([0,0,1]) #North => used to measure the dip direction
        zenith = np.asarray ([0,-1,0]) #Zenith => used to measure the dip angle
        normal = np.asarray ([x_v3, y_v3, z_v3]) #Normal of the fitting plane
        normal = normal*(-1.0 if normal[1]>=0 else 1.0) #Fix the normal if pointing in the opposite direction
        normalh = np.asarray ([normal[0], 0.0, normal[2]]) #Horizontal projection of the normal vector
        #Calculate the angle between vectors
        v1_u = normal/np.linalg.norm(normal) #For the dip_angle
        v2_u = zenith/np.linalg.norm(zenith) #For the dip_angle
        v3_u = normalh/np.linalg.norm(normalh) #For the dip_direction
        v4_u = north/np.linalg.norm(north) #For the dip_direction
        da = round(np.degrees(np.arccos(np.clip(np.dot(v1_u, v2_u), -1.0, 1.0))),0)
        dd = round(np.degrees(np.arccos(np.clip(np.dot(v3_u, v4_u), -1.0, 1.0))),0) #dip_direction
        if normalh[0]<0 and dd<=180: #Correction if it measures the complementary angle for the dip_direction
            dd = 360-dd
        elif normalh[0]>0 and dd>180: #Correction if it measures the complementary angle for the dip_direction
            dd = 360-dd
        else:
            pass
        if math.isnan(da) or math.isnan(dd):
            orient3d.append([-1.,-1.])
        else:
            orient3d.append([da, dd])
    orient3d[0][1]=orient2d[0]
    return orient3d[0]
#
##########################################################################################################
def or_2d(arr):
    """Calculate the orientation of the 2D label using SVD"""
    orient=[]
    if np.count_nonzero(arr)<2: #Check if there are less than 3 points
        orient.append(-1.)
        pass
    else:
        y, x = np.where(arr>0)
        north = [1,0]
        x = x - np.mean(x)
        y = y - np.mean(y); y = y[::-1]
        _, _, vv = np.linalg.svd(np.concatenate((x[:, np.newaxis], y[:, np.newaxis]), axis=1))
        v = vv[0]
        v = v*(-1.0 if v[1]>=0.0 else 1.0)
        v1_u = north/np.linalg.norm(north)
        v2_u = v/np.linalg.norm(v)
        ang = round(np.degrees(np.arccos(np.clip(np.dot(v1_u, v2_u), -1.0, 1.0))),0)
        if v[0]>0 and ang>90.:
            ang=360-ang
        elif v[0]<0 and ang>180:
            ang=360-ang
        orient.append(ang)
    return orient[0]
#
#vox_dim: size of the 3D local crop for orientation analysis
#matrix: mean matrix CT-value
#air: mean air CT-value
#res: FWHM value will only be measured for structures with an Edge Response (ER)>res and if the FTP is below the FWHM baseline value [((mtrx-air)/2)+air]
#z_corr: for cubic voxel is equal to 1 (put None in that case), for non cubic voxel (with x=y) is z/x
#path = r"D:\\Collab.s\\MARCEL\TEST\\Result\\3350"
#path_0 = "C:\\Users\\marce\\Desktop\\Fracture-Trace-Point-Analysis-master\\results\\"
path_0= "C:\\Users\\marce\\Desktop\\apertures_laserscans_francesco_script\\Fle_8_S1\\results\\"

fldrs = os.listdir(path_0)
for fldr in fldrs:
    #analyze_image_stack (path_0+fldr, vox_dim=11, res=200, z_corr=None)
    analyze_image_stack (path_0+fldr, vox_dim=11, res=50, z_corr=None)
```

Figure 48. Python script to measure the apertures of segmented fractures (Step1). Written by Dr. Francesco Cappuccio, the original code is described in Cappuccio et al. 2020.

```
### Dr. Francesco Cappuccio adaped his original code to measure the aperture of segmented fractures.
### His original code is published in Cappuccio F, Toy VG, Mills S, Adam L (2020) https://doi.org/10.3389/feart.2020.529263
### To apply the code you have to run the following scripts: Step0, Step1 and finally Step2
### EDIT path_0 to the results folder (used in the previous scripts, Step0 and Step1)
import os
import numpy as np
from skimage import io
from tqdm import tqdm
import matplotlib.pyplot as plt #added for histogram calculation
n_bins= 20 #put the number of bins in the histogram
def s():
    import sys
    sys.exit()
def ftp_orientation_cpu(arr):
    """It measure the orientation of the plane that best fits the FTPs"""
    import math
    np.seterr(invalid='ignore');
    arr = np.rot90(arr, axes=(0,1))
    orient3d=[]
    if np.count_nonzero(arr)<3: #Check if there are less than 3 points
        orient3d.append([-1.])
        pass
    else:
        indx = np.nonzero(arr) #Indexes of the non-zero points
        z = indx[0]
        y = indx[1]
        x = indx[2]
        x = x - np.mean(x)
        y = y - np.mean(y);# y = y[::-1]
        z = z - np.mean (z); #To center the cloud of points
        evals, evecs = np.linalg.eig(np.cov([x, y, z])) #Calculate eigenvalues and eigenvectors of the covariance matrix
        sort_indices = np.argsort(evals)[::-1]
        x_v3, y_v3, z_v3 = evecs[:, sort_indices[2]]
        north = np.asarray ([0,0,1]) #North => used to measure the dip direction
        zenith = np.asarray ([0,-1,0]) #Zenith => used to measure the dip angle
        normal = np.asarray ([x_v3, y_v3, z_v3]) #Normal of the fitting plane
        normal = normal*(-1.0 if normal[1]>=0 else 1.0) #Fix the normal if pointing in the opposite direction
        normalh = np.asarray ([normal[0], 0.0, normal[2]]) #Horizontal projection of the normal vector
        #Calculate the angle between vectors
        v1_u = normal/np.linalg.norm(normal) #For the dip angle
        v2_u = zenith/np.linalg.norm(zenith) #For the dip_angle
        v3_u = normalh/np.linalg.norm(normalh) #For the dip_direction
        v4_u = north/np.linalg.norm(north) #For the dip direction
        da = round(np.degrees(np.arccos(np.clip(np.dot(v1_u, v2_u), -1.0, 1.0))),0)
        dd = round(np.degrees(np.arccos(np.clip(np.dot(v3_u, v4_u), -1.0, 1.0))),0) #dip direction
        if normalh[0]<0 and dd<=180: #Correction if it measures the complementary angle for the dip_direction
            dd = 360-dd
        elif normalh[0]>0 and dd>180: #Correction if it measures the complementary angle for the dip_direction
            dd = 360-dd
        else:
            pass
        if math.isnan(da) or math.isnan(dd):
            orient3d.append([-1.,-1.])
        else:
            orient3d.append([da, dd])
    return orient3d[0]
##################################################################################################
#path= r"D:\\Collab.s\\MARCEL\TEST\\Result\\3350"
#os.chdir(path+r"\Minima_Analysis")

#path_0 = "C:\\Users\\marce\\Desktop\\Fracture-Trace-Point-Analysis-master\\results\\"
path_0=
```



```
"C:\\Users\\marce\\Desktop\\apertures_laserscans_francesco_script\\Fle_8_S1\\results\\
"

fldrs = os.listdir(path_0)
for fldr in fldrs:
    os.chdir(path_0+fldr+r"\\Minima_Analysis")
    img3d = io.imread("img_3dHV.tif")
    print (ftp_orientation_cpu(img3d))
    shp = img3d.shape
    img3d = np.zeros((shp), dtype='uint16')
    img3d_or = np.zeros((shp), dtype='uint8')

    ##################################################################################
    ###########################
    v = np.loadtxt("mfps_V.txt", dtype="int16", comments='#', delimiter=',')
    print (int(np.mean(v[:,4])),int(np.mean(v[:,3])))
    ap=[] #ap_app, ap_2d, ap_3d
    wrong2d=[]
    wrong3d=[]
    m_or = np.mean(v[:,3])
    for i in tqdm(range(0, len(v), 1), desc='Aperture V', ncols=100):
        ap_app = v[i,6].astype(float)
        ang2d = v[i,4]
        ang3d = v[i,3]
        if 180>ang2d>0:
            ang2d_c = ang2d if ang2d<=90 else 180-ang2d
            ap_2d = ap_app*np.cos(np.deg2rad(ang2d_c))
        elif ang2d==90 or ang2d==0 or ang2d==180:
            ap_2d = ap_app
        else:
            ap_2d = ap_app
            wrong2d.append(1)
        if ang3d>=0:
            if ang3d>0:
                ap_3d = ap_2d*np.sin(np.deg2rad(ang3d))
            else:
                ap_3d = ap_2d*np.sin(np.deg2rad(1))
        else:
            ap_3d=ap_app
            wrong3d.append(1)
        ap_3d = int(round(ap_3d))
        img3d[v[i,0],v[i,1],v[i,2]]=ap_3d
        img3d_or[v[i,0],v[i,1],v[i,2]]=int(round(np.abs(v[i,3]-m_or)))
        ap.append([ap_app, int(round(ap_2d)), ap_3d])
    io.imsave("img_ap_V.tif", img3d)
    io.imsave("img_ap_V_or.tif", img3d_or)
    del img3d; del img3d_or
    ap = np.array(ap, dtype=int, copy=False)

#plot histogram
    histogram = plt.hist(ap[:,2], bins=n_bins)


    print ('\n')
    print (path_0+fldr)
#    print (np.amax(ap[:,0]),
np.amin(ap[:,0]),int(np.mean(ap[:,0])),int(np.std(ap[:,0])))
#    print (np.amax(ap[:,1]),
np.amin(ap[:,1]),int(np.mean(ap[:,1])),int(np.std(ap[:,1])))
    print ("Max:\t{}\nMin:\t{}\nMean:\t{}\nSTD:\t{}\nRoughness:\t{}\nFTPs:\t{}".format
    (np.amax(ap[:,2]), np.amin(ap[:,2]),int(np.mean(ap[:,2])),int(np.std(ap[:,2])),
    round(np.std(ap[:,2])/np.mean(ap[:,2]),3),len(ap)))
#    print (len(ap),len(wrong2d), len(wrong3d))
#    print (int(round(m_or)), int(round(np.std(v[:,3]))))
    print ('\n')

    f= open("fracture_results.txt","w+")
    f.write("Max:\t{}\nMin:\t{}\nMean:\t{}\nSTD:\t{}\nRoughness:\t{}\nFTPs:\t{}".
    format(np.amax(ap[:,2]), np.amin(ap[:,2]),int(np.mean(ap[:,2])),int(np.std(ap[:,2
    ])),round(np.std(ap[:,2])/np.mean(ap[:,2]),3),len(ap)))
    f.close()

```

```
    #write apertures in file to create histogram in excel
    f= open("aperture_values.txt","w+")
    counter = 0;
    while counter < len(ap[:,2]):
        f.write(str(ap[counter,2])+"\n")
        counter = counter+1;
    f.close()
    del ap
```

Figure 49. Python script to measure the apertures of segmented fractures (Step2). Written by Dr. Francesco Cappuccio, the original code is described in Cappuccio et al. 2020.

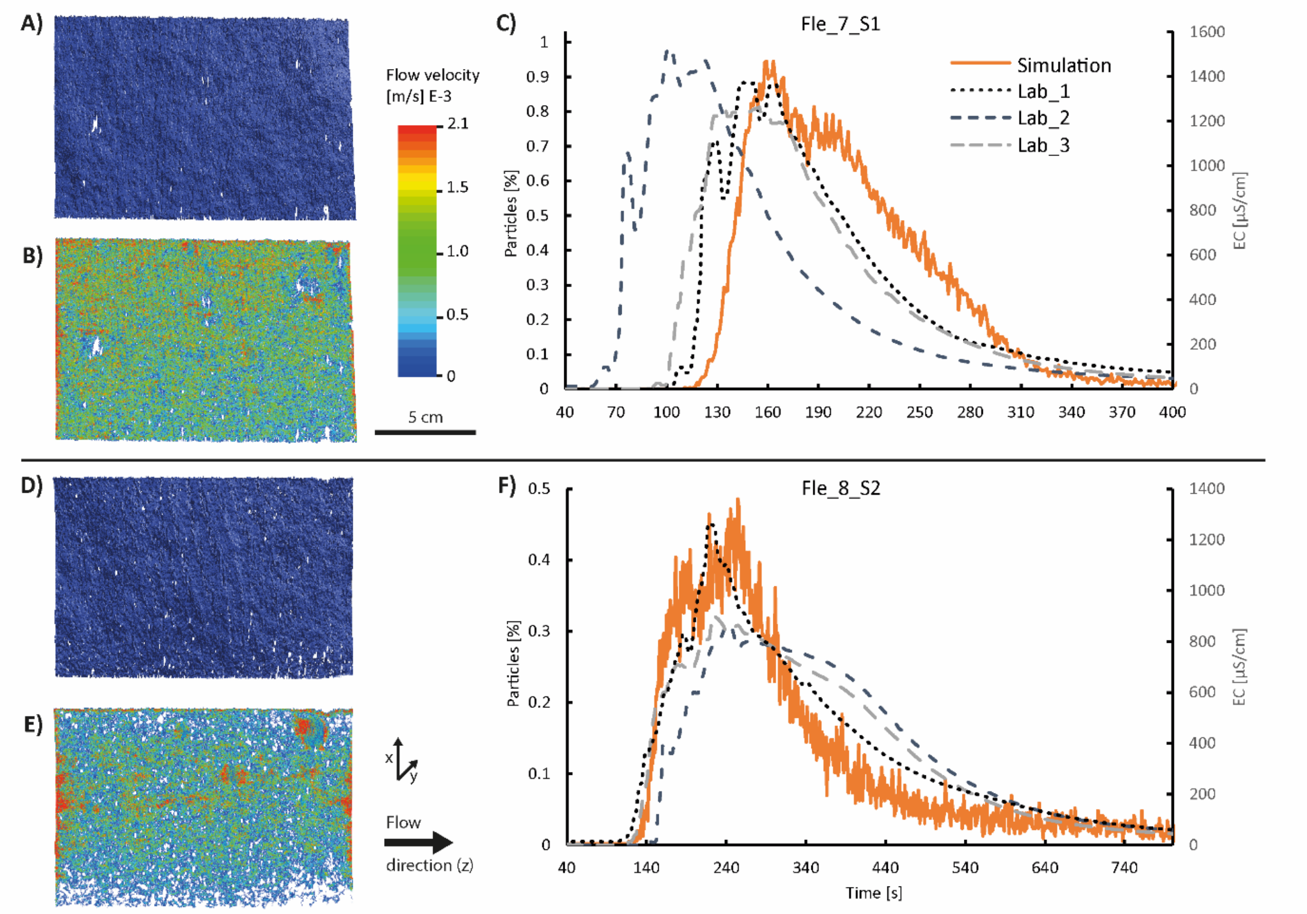


Figure 50. Result of the reconstruction of the samples Fle_7_S1 and Fle_8_S2
*The structures **(A & D)**, calculated flow fields **(B & E)**, and the simulated breakthrough curves (orange curves in **C, F**) are shown. As a comparison, for both samples, the breakthrough curves obtained by laboratory experiments are also presented in black.*

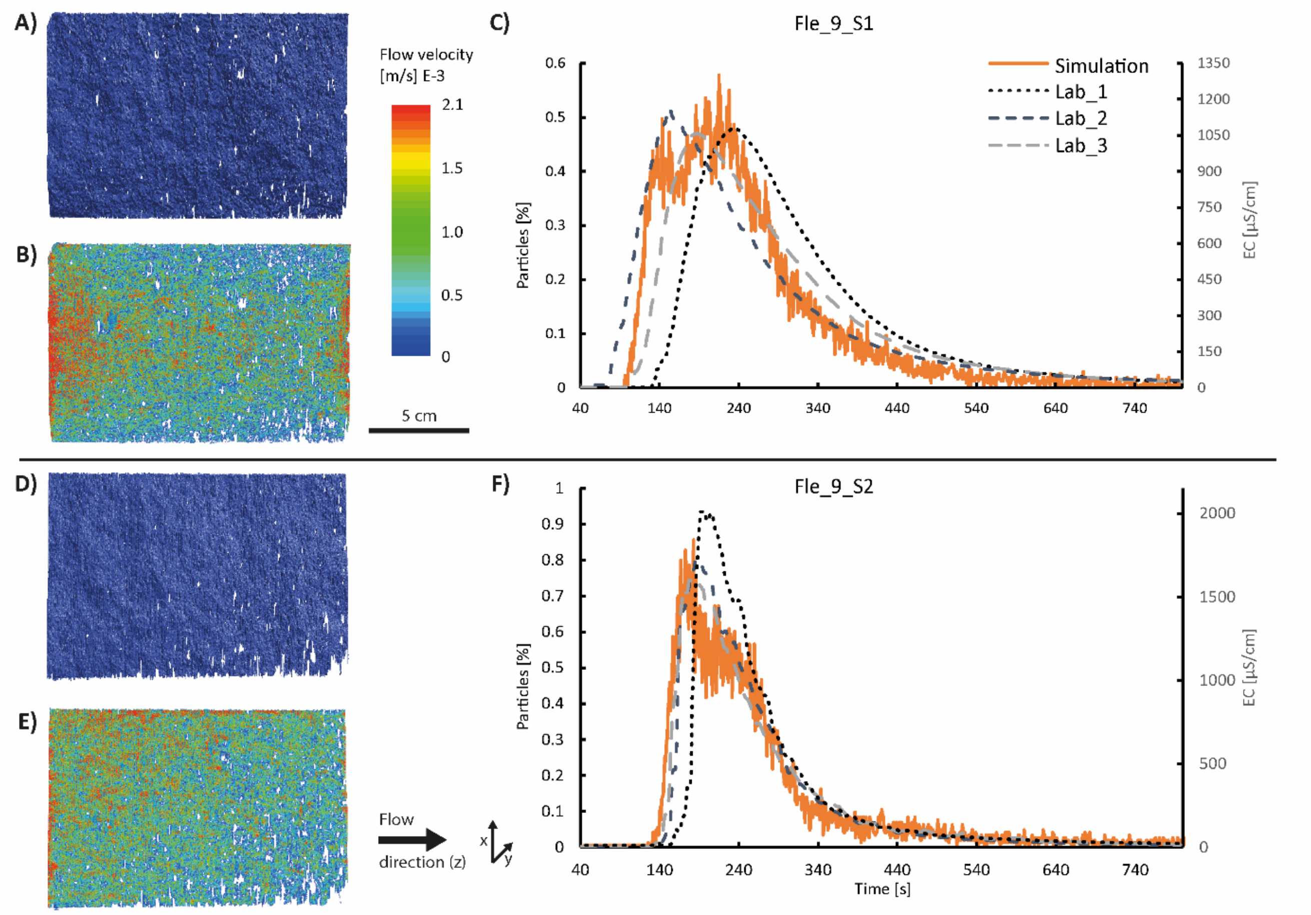


Figure 51. Result of the reconstruction of the samples Fle_9_S1 and Fle_9_S2
*The structures **(A & D)**, calculated flow fields **(B & E)**, and the simulated breakthrough curves (orange curves in **C, F**) are shown. As a comparison, for both samples, the breakthrough curves obtained by laboratory experiments are also presented in black.*

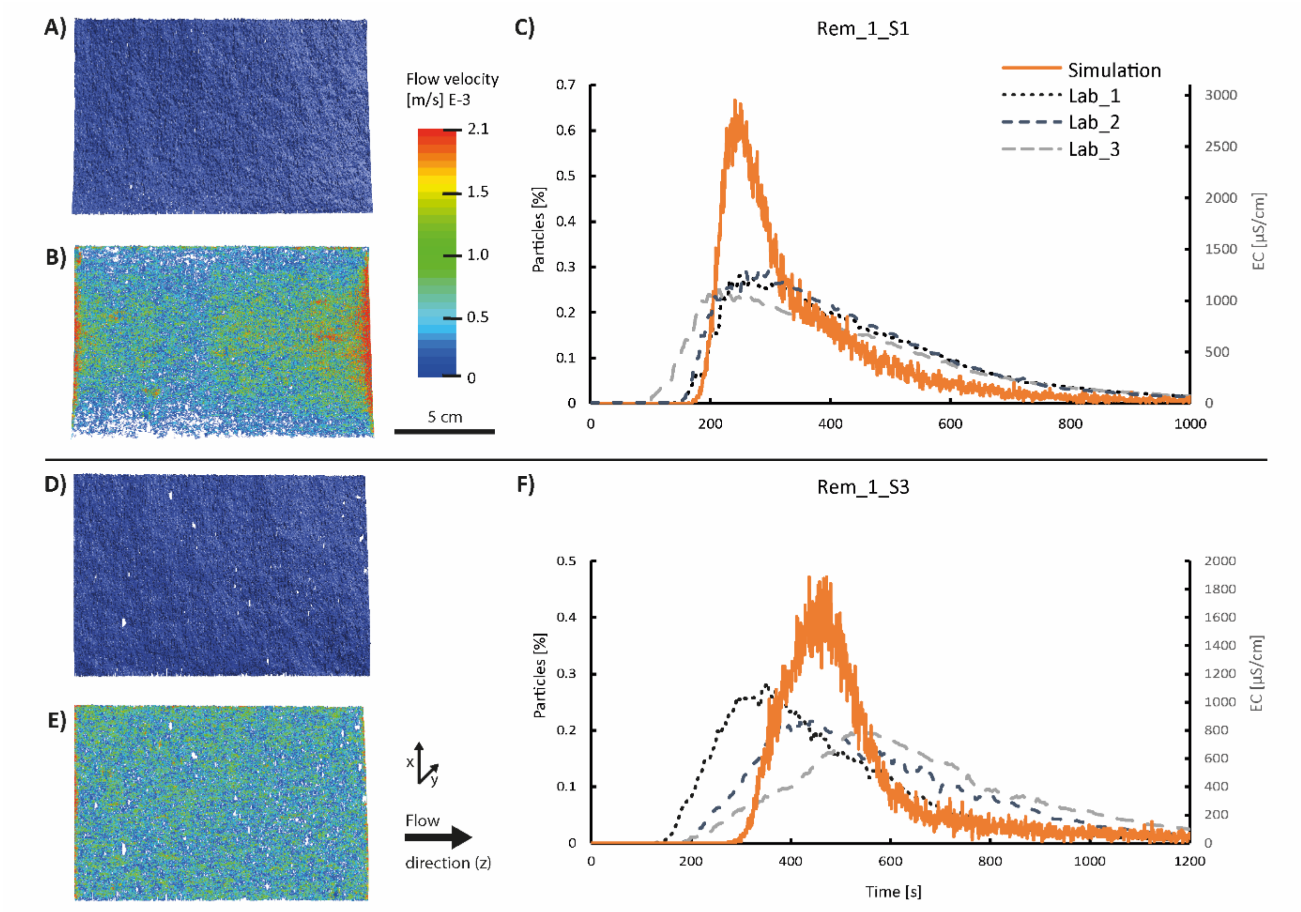


Figure 52. Result of the reconstruction of the samples Rem_1_S1 and Rem_1_S3
*The structures **(A & D)**, calculated flow fields **(B & E),** and the simulated breakthrough curves (orange curves in **C, F**) are shown. As a comparison, for both samples, the breakthrough curves obtained by laboratory experiments are also presented in black.*

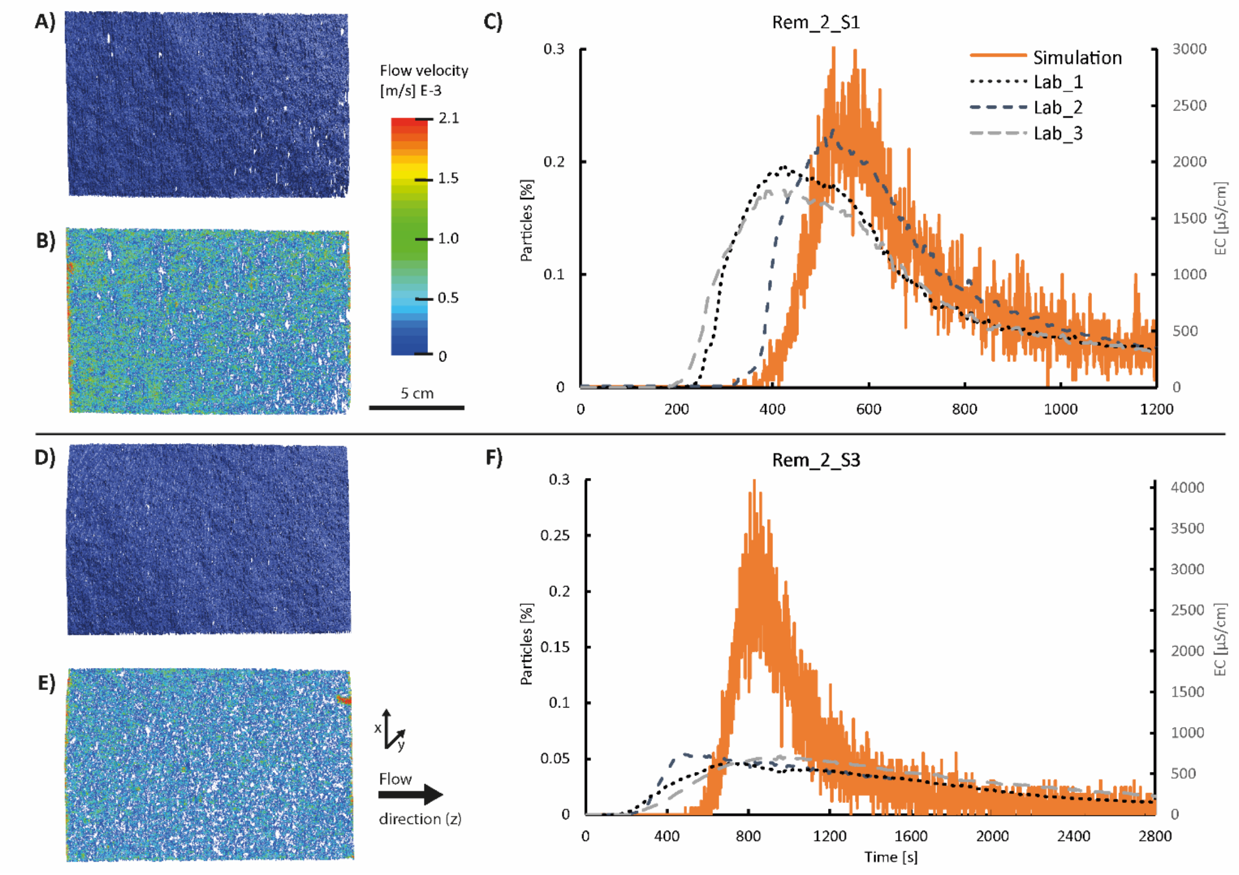


Figure 53. Result of the reconstruction of the samples Rem_2_S1 and Rem_2_S3
*The structures **(A & D)**, calculated flow fields **(B & E),** and the simulated breakthrough curves (orange curves in **C, F**) are shown. As a comparison, for both samples, the breakthrough curves obtained by laboratory experiments are also presented in black.*

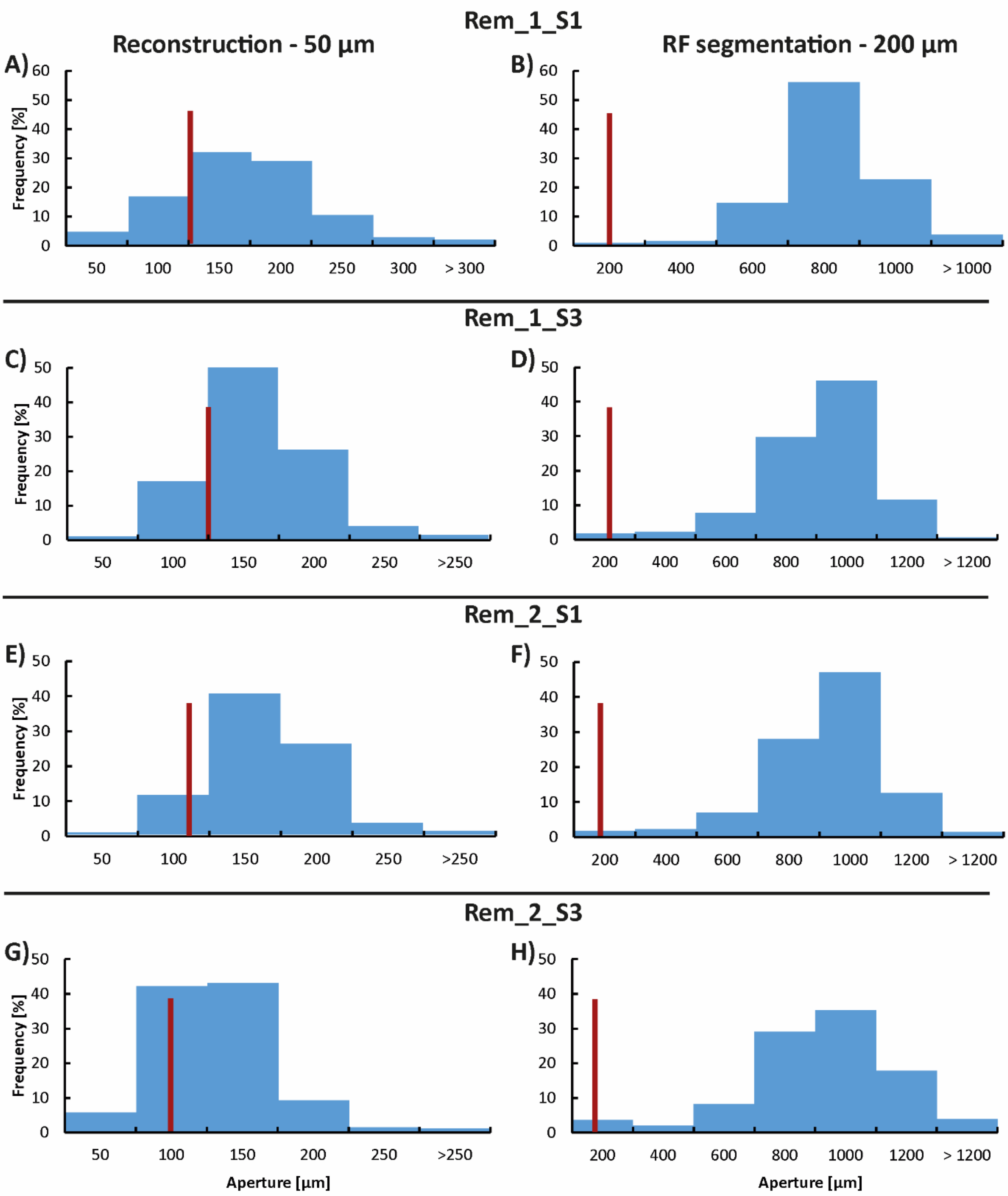


Figure 54. Apertures of the reconstructions and segmentations of the Remlinger samples
Comparison of the mechanical aperture distribution between the reconstructed fractures (with 50 µm spatial resolution, ***A, C, E, G****) and the segmented fractures (with 200 µm spatial resolution,* ***B, D, F, H****) using the RF classifier. The red line indicates the hydraulic aperture, calculated from experimental data. For the histogram classes, the bin size was chosen to be equal to the corresponding spatial voxel resolution.*

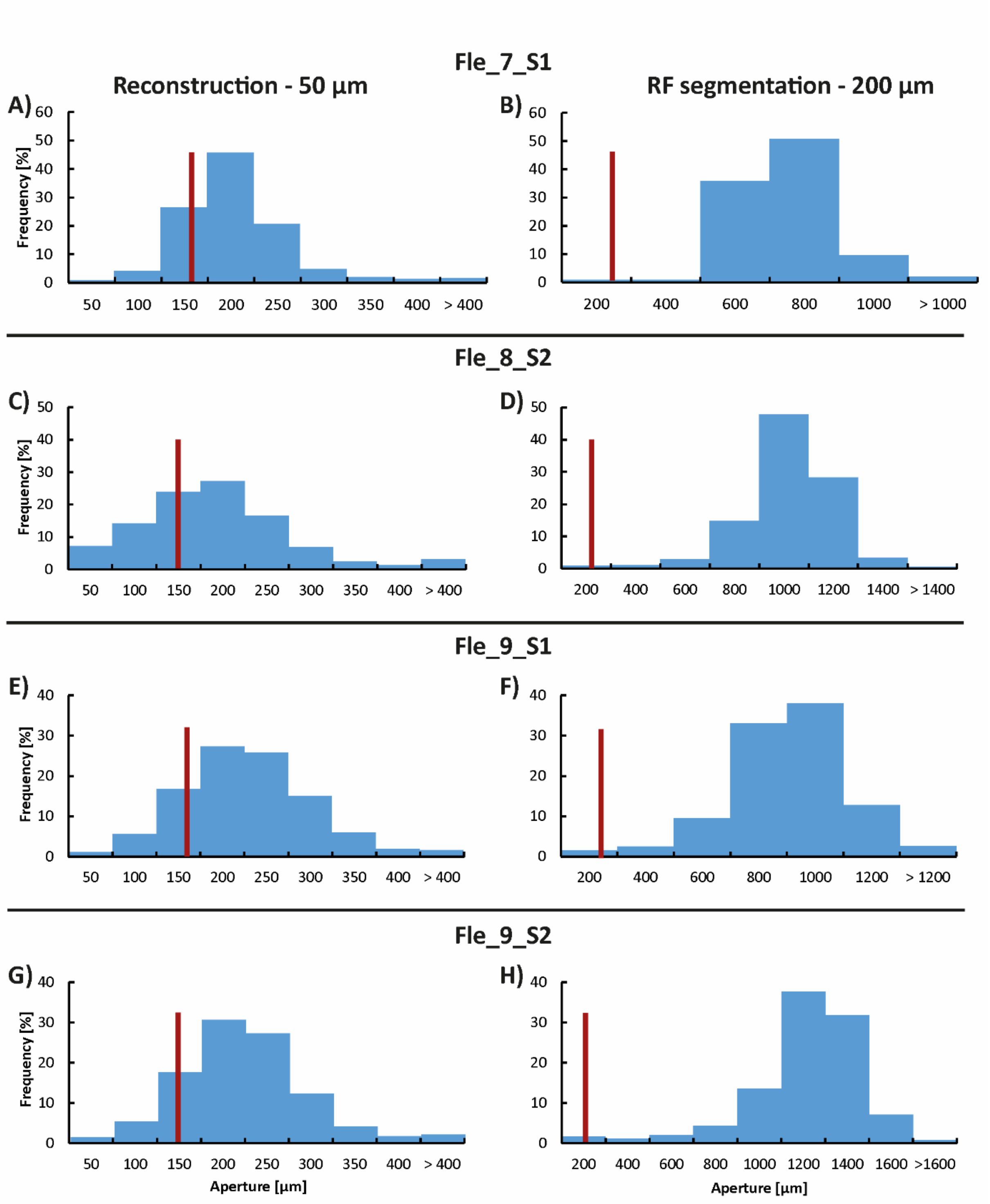


Figure 55. Apertures of the reconstructions and segmentations of the Flechtinger samples
Comparison of the mechanical aperture distribution between the reconstructed fractures (with 50 µm spatial resolution, ***A, C, E, G****) and the segmented fractures (with 200 µm spatial resolution,* ***B, D, F, H****) using the RF classifier. The red line indicates the hydraulic aperture, calculated from experimental data. For the histogram classes, the bin size was chosen to be equal to the corresponding spatial voxel resolution.*

References

Abe S, Deckert H (2021) Roughness of fracture surfaces in numerical models and laboratory experiments. Solid Earth 12:2407–2424. https://doi.org/10.5194/se-12-2407-2021

Acksel D, Amann F, Bremer J, et al (2022) Roadmap Tiefe Geothermie für Deutschland. https://doi.org/10.24406/ieg-n-645792

Agemar T, Weber J, Schulz R (2014) Deep geothermal energy production in Germany. Energies 7:4397–4416. https://doi.org/10.3390/en7074397

Ahuja VR, Gupta U, Rapole SR, et al (2022) Siamese-SR: A Siamese Super-Resolution Model for Boosting Resolution of Digital Rock Images for Improved Petrophysical Property Estimation. IEEE Trans Image Process 31:3479–3493. https://doi.org/10.1109/TIP.2022.3172211

Akin S, Kovscek AR (2003) Computed tomography in petroleum engineering research. Geol Soc London, Spec Publ 215:23–38. https://doi.org/10.1144/GSL.SP.2003.215.01.03

Akyol E, Aras Ö, Öner M (2014) Control of barium sulfate crystallization in the presence of additives. Desalin Water Treat 52:5965–5973. https://doi.org/10.1080/19443994.2013.808589

Al-Marzouqi H (2018) Digital Rock Physics: Using CT Scans to Compute Rock Properties. IEEE Signal Process Mag 35:121–131. https://doi.org/10.1109/MSP.2017.2784459

Alaskar M, Li K, Horne R (2015) Silica Particles Mobility Through Fractured Rock. Arab J Sci Eng 40:1205–1222. https://doi.org/10.1007/s13369-015-1577-1

Alqahtani NJ, Niu Y, Wang Y Da, et al (2022) Super-Resolved Segmentation of X-ray Images of Carbonate Rocks Using Deep Learning. Transp Porous Media 143:497–525. https://doi.org/10.1007/s11242-022-01781-9

Andrä H, Combaret N, Dvorkin J, et al (2013a) Digital rock physics benchmarks—Part I: Imaging and segmentation. Comput Geosci 50:25–32. https://doi.org/10.1016/j.cageo.2012.09.005

Andrä H, Combaret N, Dvorkin J, et al (2013b) Digital rock physics benchmarks—part II: Computing effective properties. Comput Geosci 50:33–43. https://doi.org/10.1016/j.cageo.2012.09.008

Andrew M (2018) A quantified study of segmentation techniques on synthetic geological XRM and FIB-SEM images. Comput Geosci 22:1503–1512. https://doi.org/10.1007/s10596-018-9768-y

Andrew M, Omlor L, Andreyev A, et al (2021) New technologies for x-ray microscopy: phase correction and fully automated deep learning based tomographic reconstruction. In: Müller B, Wang G (eds) Developments in X-Ray Tomography XIII. SPIE, p 15

Arganda-Carreras I, Kaynig V, Rueden C, et al (2017) Trainable Weka Segmentation: A machine learning tool for microscopy pixel classification. Bioinformatics 33:2424–2426. https://doi.org/10.1093/bioinformatics/btx180

Babel H, Naranjo-Meneses P, Trauth S, et al (2020) Ratiometric population sensing by a pump-probe signaling system in Bacillus subtilis. Nat Commun 11:1176. https://doi.org/10.1038/s41467-020-14840-w

Badrinarayanan V, Kendall A, Cipolla R (2017) SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation. IEEE Trans Pattern Anal Mach Intell 39:2481–2495. https://doi.org/10.1109/TPAMI.2016.2644615

Baghaie A, Yu Z (2015) Structure tensor based image interpolation method. AEU - Int J Electron Commun 69:515–522. https://doi.org/10.1016/j.aeue.2014.10.022

Barbier E (2002) Geothermal energy technology and current status: an overview. Renew Sustain Energy Rev 6:3–65. https://doi.org/10.1016/S1364-0321(02)00002-3

Bandis SC, Lumsden AC, Barton NR (1983) Fundamentals of rock joint deformation. Int J Rock Mech Min Sci Geomech Abstr 20:249–268. https://doi.org/10.1016/0148-9062(83)90595-8

Baraboshkin EE, Ismailova LS, Orlov DM, et al (2020) Deep convolutions for in-depth automated rock typing. Comput Geosci 135:104330. https://doi.org/10.1016/j.cageo.2019.104330

Barton N (1973) Review of a new shear-strength criterion for rock joints. Eng Geol 7:287–332. https://doi.org/10.1016/0013-7952(73)90013-6

Barton N, Bandis S, Bakhtar K (1985) Strength, deformation and conductivity coupling of rock joints. Int J Rock Mech Min Sci 22:121–140. https://doi.org/10.1016/0148-9062(85)93227-9

Barton N, de Quadros EF (1997) Joint aperture and roughness in the prediction of flow and groutability of rock masses. Int J Rock Mech Min Sci 34:252.e1-252.e14. https://doi.org/10.1016/S1365-1609(97)00081-6

Batchelor GK (2000) An Introduction to Fluid Dynamics. Cambridge University Press, Cambridge.

Baticci F, Genter A, Huttenloch P, Zorn R (2010) Corrosion and scaling detection in the Soultz EGS power plant, Upper Rhine graben, France. In: Proceedings World Geothermal Congress, Bali, Indonesia, 25-29 April, p. 11.

Bauget F, Fourar M (2008) Non-Fickian dispersion in a single fracture. J Contam Hydrol 100:137–148. https://doi.org/10.1016/j.jconhyd.2008.06.005

Becker J, Hilden J, Rief S, Planas B (2020) PoroDict user guide, GeoDict 2021 from Math2Market GmbH, Germany, https://doi.org/10.30423/userguide.geodict2021-porodict (accessed 20.01.2022).

Becker TA (2008) Zur Strömungssimulation in Einzelklüften: Gegenüberstellung von numerischen Methoden und Experiment; Dissertation; Rheinisch-Westfälische Technische Hochschule Aachen: Aachen, Germany

Becker MW, Reimus PW, Vilks P (1999) Transport and Attenuation of Carboxylate-Modified Latex Microspheres in Fractured Rock Laboratory and Field Tracer Tests. Ground Water 37:387–395. https://doi.org/10.1111/j.1745-6584.1999.tb01116.x

Berg S, Kutra D, Kroeger T, et al (2019) Ilastik: Interactive Machine Learning for (Bio)Image Analysis. Nat Methods 16:1226–1232. https://doi.org/10.1038/s41592-019-0582-9

Berg S, Saxena N, Shaik M, Pradhan C (2018) Generation of ground truth images to validate micro-CT image-processing pipelines. Lead Edge 37:412–420. https://doi.org/10.1190/tle37060412.1

Berkowitz B, Scher H (2009) Exploring the nature of non-Fickian transport in laboratory experiments. Adv Water Resour 32:750–755. https://doi.org/10.1016/j.advwatres.2008.05.004

Bertram MA, Cowen JP (1997) Morphological and compositional evidence for biotic precipitation of marine barite. J Mar Res 55:577–593. https://doi.org/10.1357/0022240973224292

Beucher S, Meyer F (1993) The morphological approach to segmentation: the watershed transformation. In: Dougherty ER (ed) Mathematical morphology in image processing, Marcel Dekker Inc., New York, pp 433-481

Biau G, Scornet E (2016) A random forest guided tour. TEST 25:197–227. https://doi.org/10.1007/s11749-016-0481-7

Bijeljic B, Muggeridge AH, Blunt MJ (2004) Pore-scale modeling of longitudinal dispersion. Water Resour Res 40:1–9. https://doi.org/10.1029/2004WR003567

Birner J, Seibt A, Hinrichs T, Seibt P, Wolfgramm M (2015) Removing and Reducing Scalings—Practical

Experience in the Operation of Geothermal Systems. In Proceedings of the World Geothermal Congress, Melbourne, Australia

Blöcher G, Kluge C, Milsch H, et al (2019) Permeability of matrix-fracture systems under mechanical loading – constraints from laboratory experiments and 3-D numerical modelling. Adv Geosci 49:95–104. https://doi.org/10.5194/adgeo-49-95-2019

Blöcher G, Reinsch T, Hassanzadegan A, et al (2014) Direct and indirect laboratory measurements of poroelastic properties of two consolidated sandstones. Int J Rock Mech Min Sci 67:191–201. https://doi.org/10.1016/j.ijrmms.2013.08.033

Blöcher G, Reinsch T, Henninges J, et al (2016) Hydraulic history and current state of the deep geothermal reservoir Groß Schönebeck. Geothermics 63:27–43. https://doi.org/10.1016/j.geothermics.2015.07.008

Blöcher G, Zimmermann G, Milsch H (2009) Impact of poroelastic response of sandstones on geothermal power production. Pure Appl Geophys 166:1107–1123. https://doi.org/10.1007/s00024-009-0475-4

Blunt MJ (2017) Multiphase Flow in Permeable Media: A Pore-Scale Perspective. Cambridge University Press, Cambridge.

BMU - Bundesministerium für Umwelt, Naturschutz und nukleare Sicherheit (2021) Klimaschutz in Zahlen Fakten, Trends und Impulse deutscher Klimapolitik Ausgabe 2021, 1st ed., BMU, Berlin. https://www.bmuv.de/publikation/klimaschutz-in-zahlen-2021 (accessed 18.02.2022).

BMWi - Bundesministerium für Wirtschaft und Energie (2021) Erneuerbare Energien in Zahlen - Nationale und internationale Entwicklung im Jahr 2020. BMWi, Berlin. https://www.bmwi.de/Redaktion/DE/Publikationen/Energie/erneuerbare-energien-in-zahlen-2020.html (accessed 18.02.2022).

Bodin J, Delay F, de Marsily G (2003a) Solute transport in a single fracture with negligible matrix permeability: 1. Fundamental mechanisms. Hydrogeol J 11:418–433. https://doi.org/10.1007/s10040-003-0268-2

Bodin J, Delay F, de Marsily G (2003b) Solute transport in a single fracture with negligible matrix permeability: 2. Mathematical formalism. Hydrogeol J 11:434–454. https://doi.org/10.1007/s10040-003-0269-1

Boutt DF, Grasselli G, Fredrich JT, et al (2006) Trapping zones: The effect of fracture roughness on the directional anisotropy of fluid flow and colloid transport in a single fracture. Geophys Res Lett 33:1–6. https://doi.org/10.1029/2006GL027275

Brady EL, Kirby MA, Olszewski E, et al (2020) Guided vascularization in the rat heart leads to transient vessel patterning. APL Bioeng 4:016105. https://doi.org/10.1063/1.5122804

Bravin A, Coan P, Suortti P (2013) X-ray phase-contrast imaging: from pre-clinical applications towards clinics. Phys Med Biol 58:R1–R35. https://doi.org/10.1088/0031-9155/58/1/R1

Breiman L (2001) Random Forest. In: Random Forest. CRC Press, First. | Boca Raton : CRC Press, 2019., pp 5–32

Briggs S, Karney BW, Sleep BE (2017) Numerical modeling of the effects of roughness on flow and eddy formation in fractures. J Rock Mech Geotech Eng 9:105–115. https://doi.org/10.1016/j.jrmge.2016.08.004

Brown SR (1995) Simple mathematical model of a rough fracture. J Geophys Res Solid Earth 100:5941–5952. https://doi.org/10.1029/94JB03262

Brush DJ, Thomson NR (2003) Fluid flow in synthetic rough-walled fractures: Navier-Stokes, Stokes, and local cubic law simulations. Water Resour Res 39:. https://doi.org/10.1029/2002WR001346

Bührer M, Xu H, Eller J, et al (2020) Unveiling water dynamics in fuel cells from time-resolved tomographic microscopy data. Sci Rep 10:16388. https://doi.org/10.1038/s41598-020-73036-w

Bultreys T, Boone MA, Boone MN, et al (2016) Fast laboratory-based micro-computed tomography for pore-scale research: Illustrative experiments and perspectives on the future. Adv Water Resour 95:341–351. https://doi.org/10.1016/j.advwatres.2015.05.012

Byun H, Kim J, Yoon D, et al (2021) A deep convolutional neural network for rock fracture image segmentation. Earth Sci Informatics 14:1937–1951. https://doi.org/10.1007/s12145-021-00650-1

Candela T, Renard F, Bouchon M, et al (2009) Characterization of Fault Roughness at Various Scales: Implications of Three-Dimensional High Resolution Topography Measurements. Pure Appl Geophys 166:1817–1851. https://doi.org/10.1007/s00024-009-0521-2

Candela T, Renard F, Klinger Y, et al (2012) Roughness of fault surfaces over nine decades of length scales. J Geophys Res Solid Earth 117:n/a-n/a. https://doi.org/10.1029/2011JB009041

Cappuccio F, Toy VG, Mills S, Adam L (2020) Three-Dimensional Separation and Characterization of Fractures in X-Ray Computed Tomographic Images of Rocks. Front Earth Sci 8:1–13. https://doi.org/10.3389/feart.2020.529263

Chagneau A, Claret F, Enzmann F, et al (2015) Mineral precipitation-induced porosity reduction and its effect on transport parameters in diffusion-controlled porous media. Geochem Trans 16:1–16. https://doi.org/10.1186/s12932-015-0027-z

Chauhan S, Rühaak W, Khan F, et al (2016) Processing of rock core microtomography images: Using seven different machine learning algorithms. Comput Geosci 86:120–128. https://doi.org/10.1016/j.cageo.2015.10.013

Chen J, Zhou L, Li H, et al (2020) Comparison on Hydraulic and Mechanical Responses to the Growing Loading between Both Fractures Induced by Hydraulic Fracturing and Brazilian Splitting. Energy and Fuels 34:10933–10946. https://doi.org/10.1021/acs.energyfuels.0c02233

Chen Z, Zhan H, Zhao G, et al (2017) Effect of Roughness on Conservative Solute Transport through Synthetic Rough Single Fractures. Water 9:656. https://doi.org/10.3390/w9090656

Cheng C, Hale S, Milsch H, Blum P (2020) Measuring hydraulic fracture apertures: A comparison of methods. Solid Earth 11:2411–2423. https://doi.org/10.5194/se-11-2411-2020

Cheng C, Milsch H (2021) Hydromechanical Investigations on the Self-propping Potential of Fractures in Tight Sandstones. Rock Mech Rock Eng 54:5407–5432. https://doi.org/10.1007/s00603-021-02500-4

Chrysikopoulos C V., Abdel-Salam A (1997) Modeling colloid transport and deposition in saturated fractures. Colloids Surfaces A Physicochem Eng Asp 121:189–202. https://doi.org/10.1016/S0927-7757(96)03979-9

Clausnitzer V, Hopmans JW (2000) Pore-scale measurements of solute breakthrough using microfocus X-ray computed tomography. Water Resour Res 36:2067–2079. https://doi.org/10.1029/2000WR900076

Clennell M Ben (1997) Tortuosity: a guide through the maze. Geol Soc London, Spec Publ 122:299–344. https://doi.org/10.1144/GSL.SP.1997.122.01.18

Cnudde V, Boone MN (2013) High-resolution X-ray computed tomography in geosciences: A review of the current technology and applications. Earth-Science Rev 123:1–17.

https://doi.org/10.1016/j.earscirev.2013.04.003

Constantin P, Iyer G (2008) A stochastic Lagrangian representation of the three-dimensional incompressible Navier-Stokes equations. Commun Pure Appl Math 61:330–345. https://doi.org/10.1002/cpa.20192

Cook AM, Myer LR, Cook NG, Doyle FM (1990) The effect of tortuosity on flow through a natural fracture. In: Hustrulid WA, Johnson GA (eds) Rock mechanics contributions and challenges: proceeding of the 31st US symposium. AA Balkema, Golden, pp 371–378

Cumbie DH, McKay LD (1999) Influence of diameter on particle transport in a fractured shale saprolite. J Contam Hydrol 37:139–157. https://doi.org/10.1016/S0169-7722(98)00156-9

Cunningham E (1910) On the Velocity of Steady Fall of Spherical Particles through Fluid Medium. Proceedings of the Royal Society of London A 83:357-365. https://doi.org/10.1098/rspa.1910.0024

Darcy H (1856) Les Fontaines Publiques de la Ville de Dijon. Dal- mont, Kraljevica

Davis G., Elliott J. (1997) X-ray microtomography scanner using time-delay integration for elimination of ring artefacts in the reconstructed image. Nucl Instruments Methods Phys Res Sect A Accel Spectrometers, Detect Assoc Equip 394:157–162. https://doi.org/10.1016/S0168-9002(97)00566-4

Dang W, Wu W, Konietzky H, Qian J (2019) Effect of shear-induced aperture evolution on fluid flow in rock fractures. Comput Geotech 114:. https://doi.org/10.1016/j.compgeo.2019.103152

Davis GR, Elliott JC (2006) Artefacts in X-ray microtomography of materials. Mater Sci Technol 22:1011–1018. https://doi.org/10.1179/174328406X114117

De Marsily G (1986) Quantitative Hydrogeology-Groundwater Hydrology for Engineers, 1st ed.; Academic Press: Orlando, FL, USA. ISBN 0-12-208915-4

Deng L, Yu D (2014) Deep Learning: Methods and Applications. Found Trends® Signal Process 7:197–387. https://doi.org/10.1561/2000000039

Ding Y, Meng X, Yang D (2021) Numerical simulation of polydisperse dense particles transport in a random-orientated fracture with spatially variable apertures. Colloids Surfaces A Physicochem Eng Asp 610:125729. https://doi.org/10.1016/j.colsurfa.2020.125729

Dippenaar MA, Van Rooy JL (2016) On the cubic law and variably saturated flow through discrete open rough-walled discontinuities. Int J Rock Mech Min Sci 89:200–211. https://doi.org/10.1016/j.ijrmms.2016.09.011

Diwakar M, Kumar M (2018) A review on CT image noise and its denoising. Biomed Signal Process Control 42:73–88. https://doi.org/10.1016/j.bspc.2018.01.010

Dou Z, Sleep B, Zhan H, et al (2019) Multiscale roughness influence on conservative solute transport in self-affine fractures. Int J Heat Mass Transf 133:606–618. https://doi.org/10.1016/j.ijheatmasstransfer.2018.12.141

Dou Z, Zhang X, Wang J, et al (2021) Influence of Grain Size Transition on Flow and Solute Transport through 3D Layered Porous Media. Lithosphere 2021:. https://doi.org/10.2113/2021/7064502/5445171/7064502.pdf

Dou Z, Zhou Z, Wang J, Huang Y (2018) Roughness scale dependence of the relationship between tracer longitudinal dispersion and Peclet number in variable-aperture fractures. Hydrol Process 32:1461–1475. https://doi.org/10.1002/hyp.11505

Einstein A (1956) Investigations on the theory of the Brownian movement. Courier, New York

El-Soudani SM (1978) Profilometric analysis of fractures. Metallography 11:247–336. https://doi.org/10.1016/0026-0800(78)90045-9

Eyerer S, Schifflechner C, Hofbauer S, et al (2020) Combined heat and power from hydrothermal geothermal resources in Germany: An assessment of the potential. Renew Sustain Energy Rev 120:109661. https://doi.org/10.1016/j.rser.2019.109661

Eyerer S, Schifflechner C, Hofbauer S, et al (2017) Potential der hydrothermalen Geothermie zur Stromerzeugung in Deutschland. Technical Report.

Eyolfson E, Yamakawa GR, Griep Y, et al (2020) Examining the Progressive Behavior and Neuropathological Outcomes Associated with Chronic Repetitive Mild Traumatic Brain Injury in Rats. Cereb Cortex Commun 1–38. https://doi.org/10.1093/texcom/tgaa002

Fardin N, Stephansson O, Jing L (2001) The scale dependence of rock joint surface roughness. Int J Rock Mech Min Sci 38:659–669. https://doi.org/10.1016/S1365-1609(01)00028-4

Feng J, He X, Teng Q, et al (2019) Reconstruction of porous media from extremely limited information using conditional generative adversarial networks. Phys Rev E 100:033308.

https://doi.org/10.1103/PhysRevE.100.033308

Ficker T (2017) Fractal properties of joint roughness coefficients. Int J Rock Mech Min Sci 94:27–31. https://doi.org/10.1016/j.ijrmms.2017.02.014

Frank S, Heinze T, Ribbers M, Wohnlich S (2020a) Experimental Reproducibility and Natural Variability of Hydraulic Transport Properties of Fractured Sandstone Samples. Geosciences 10:458. https://doi.org/10.3390/geosciences10110458

Frank S, Heinze T, Wohnlich S (2020b) Comparison of surface roughness and transport processes of sawed, split and natural sandstone fractures. Water (Switzerland) 12:. https://doi.org/10.3390/w12092530

Frank S, Zuber P, Pollak S, et al (2021) A High-Pressure High-Temperature Column for the Simulation of Hydrothermal Water Circulation at Laboratory Scale. Geotech Test J 44:. https://doi.org/10.1520/GTJ20200020

Frick S, Regenspurg S, Kranz S, et al (2011) Geochemical and Process Engineering Challenges for Geothermal Power Generation. Chemie Ing Tech 83:2093–2104. https://doi.org/10.1002/cite.201100131

Garfi G, John CM, Berg S, Krevor S (2020) The Sensitivity of Estimates of Multiphase Fluid and Solid Properties of Porous Rocks to Image Processing. Transp Porous Media 131:985–1005. https://doi.org/10.1007/s11242-019-01374-z

Ge S (1997) A governing equation for fluid flow in rough fractures. Water Resour Res 33:53–61. https://doi.org/10.1029/96WR02588

Glatt E, Huber F, Enzmann F, Schaefer T, Wiegmann A (2015) GeoDict Case Study –Nanoparticle migration in a natural granite fracture. Math2Market GmbH, Kaiserslautern. https://www.geodict.com/Showroom/CaseStudies/M2MNanoParticleMigration_CaseStudy.pdf (accessed 20.01.2022).

Goldfarb E, Ikeda K, Tisato N (2017) Segmentationless digital rock physics using different effective medium theories. In: SEG Technical Program Expanded Abstracts 2017. Society of Exploration Geophysicists, pp 3908–3913

Goldfarb EJ, Ikeda K, Ketcham RA, et al (2022) Predictive digital rock physics without segmentation. Comput Geosci 159:105008. https://doi.org/10.1016/j.cageo.2021.105008

Gonzales RC, Woods RE (2007) Digital Image Processing (Third Edition). Prentice Hall International, Upper Saddle River.

Goodfellow I, Bengio Y, Courville A (2016) Deep Learning. MIT Press, Cambridge.

Griffiths L, Heap MJ, Wang F, et al (2016) Geothermal implications for fracture-filling hydrothermal precipitation. Geothermics 64:235–245. https://doi.org/10.1016/j.geothermics.2016.06.006

Guntoro PI, Ghorbani Y, Koch P-H, Rosenkranz J (2019) X-ray Microcomputed Tomography (µCT) for Mineral Characterization: A Review of Data Analysis Methods. Minerals 9:183. https://doi.org/10.3390/min9030183

Guo Y, Liu Y, Oerlemans A, et al (2016) Deep learning for visual understanding: A review. Neurocomputing 187:27–48. https://doi.org/10.1016/j.neucom.2015.09.116

Gupta G, Chandel R (2013) Image Filtering Algorithms and Techniques: A Review. Int J Adv Res Comput Sci Softw Eng 3:2277

Gutjahr T, Hale S, Keller K, et al (2021) Quantification of Fracture Roughness by Change Probabilities and Hurst Exponents. Math Geosci. https://doi.org/10.1007/s11004-021-09985-3

Haas-Nüesch R, Heberling F, Schild D, et al (2018) Mineralogical characterization of scalings formed in geothermal sites in the Upper Rhine Graben before and after the application of sulfate inhibitors. Geothermics 71:264–273. https://doi.org/10.1016/j.geothermics.2017.10.006

Hale S, Naab C, Butscher C, Blum P (2020) Method Comparison to Determine Hydraulic Apertures of Natural Fractures. Rock Mech Rock Eng 53:1467–1476. https://doi.org/10.1007/s00603-019-01966-7

Hale S, Trindade Pedrosa E, Jacob A, et al (2022) Upscaling calcite dissolution rates in a tight reservoir sandstone. Environ Earth Sci 81:303. https://doi.org/10.1007/s12665-022-10399-5

Hamaker HC (1937) The London—van der Waals attraction between spherical particles. physica 4, pp 1058-1072

Hassannayebi N, Jammernegg B, Schritter J, et al (2021) Relationship Between Microbial Growth and Hydraulic Properties at the Sub-Pore Scale. Transp Porous Media 139:579–593. https://doi.org/10.1007/s11242-021-01680-5

Hassanzadegan A, Blöcher G, Zimmermann G, Milsch H (2012) Thermoporoelastic properties of

Flechtinger sandstone. Int J Rock Mech Min Sci 49:94–104. https://doi.org/10.1016/j.ijrmms.2011.11.002

He X, Sinan M, Kwak H, Hoteit H (2021) A corrected cubic law for single-phase laminar flow through rough-walled fractures. Adv Water Resour 154:103984. https://doi.org/10.1016/j.advwatres.2021.103984

Heiland J (2003) Permeability of triaxially compressed sandstone: Influence of deformation and strain-rate on permeability. Pure Appl Geophys 160:889–908. https://doi.org/10.1007/PL00012571

Hennessy AJB, Graham GM (2002) The effect of additives on the co-crystallisation of calcium with barium sulphate. J Cryst Growth 237–239:2153–2159. https://doi.org/10.1016/S0022-0248(01)02258-8

Hilden J, Linden S, Planas B (2020) FlowDict user guide, GeoDict 2021 from Math2Market GmbH, Germany, https://doi.org/10.30423/userguide.geodict2021-flowdict (accessed 20.01.2022).

Hinton G, Srivastava N, Swersky K (2012) Neural networks for machine learning lecture 6a overview of mini-batch gradient descent. Lect. Slides

Hinz C (2019) Reactive flow in porous media based on numerical simulations at the pore scale. Dissertation, Johannes-Gutenberg Universität Mainz: Mainz, Germany. http://doi.org/10.25358/openscience-3102

Hinz C, Enzmann F, Kersten M (2019) Pore scale modelling of calcite cement dissolution in a reservoir sandstone matrix. E3S Web Conf 98:05010. https://doi.org/10.1051/e3sconf/20199805010

Hiramatsu Y, Hotta K, Imanishi A, et al (2018) Cell Image Segmentation by Integrating Multiple CNNs. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW). IEEE, pp 2286–22866

Hölting B, Coldewey WG (2005) Hydrogeologie. Einführung in die Allgemeine und Angewandte Hydrogeologie. Elsevier GmbH, München, 6th edition, 326 p.

Holz M, Heil SR, Sacco A (2000) Temperature-dependent self-diffusion coefficients of water and six selected molecular liquids for calibration in accurate 1H NMR PFG measurements. Phys Chem Chem Phys 2:4740–4742. https://doi.org/10.1039/b005319h

Hong J, Liu J (2020) Rapid estimation of permeability from digital rock using 3D convolutional neural network. Comput Geosci 24:1523–1539. https://doi.org/10.1007/s10596-020-09941-w

Hounsfield GN (1980) Nobel lecture, 8, December 1979. Computed medical imaging. Journal of Radiology 61:459–468

Hsiao PY, Chou SS, Huang FC (2007) Generic 2-D gaussian smoothing filter for noisy image processing. In: TENCON 2007 - 2007 IEEE Region 10 Conference. IEEE, pp 1–4

Hu Y, Xu W, Zhan L, et al (2020) Non-Fickian Solute Transport in Rough-Walled Fractures: The Effect of Contact Area. Water 12:2049. https://doi.org/10.3390/w12072049

Huo D, Pini R, Benson SM (2016) A calibration-free approach for measuring fracture aperture distributions using X-ray computed tomography. Geosphere 12:558–571. https://doi.org/10.1130/GES01175.1

Huttrer GW (2020) Geothermal power generation in the world 2015-2020 update report Proc. World Geotherm. Congr. Reykjavik, 2020, Iceland, pp. 1-17

Iassonov P, Gebrenegus T, Tuller M (2009) Segmentation of X-ray computed tomography images of porous materials: A crucial step for characterization and quantitative analysis of pore structures. Water Resour Res 45:. https://doi.org/10.1029/2009WR008087

Ikeda K, Goldfarb EJ, Tisato N (2020) Calculating Effective Elastic Properties of Berea Sandstone Using the Segmentation-Less Method Without Targets. J Geophys Res Solid Earth 125:1–19. https://doi.org/10.1029/2019JB018680

Ilzig T, Günther S, Odenbach S (2022) Combined beam hardening artifact correction and quantitative microanalysis of colloidal depositions in deep bed filtration experiments investigated by 3D X-ray computed microtomography. Micron 158:103265. https://doi.org/10.1016/j.micron.2022.103265

Ioffe S, Szegedy C (2015) Batch normalization: Accelerating deep network training by reducing internal covariate shift. 32nd Int Conf Mach Learn ICML 2015 1:448–456

IPCC (2014) Climate Change 2014: Mitigation of Climate Change. Contribution of Working Group III to the Fifth Assessment Report of the Intergovernmental Panel on Climate Change [Edenhofer, O., R. Pichs-Madruga, Y. Sokona, E. Farahani, S. Kadner, K. Seyboth, A. Adler, I. Baum, S. Brunner, P. Eickemeier, B. Kriemann, J. Savolainen, S. Schlömer, C. von Stechow, T. Zwickel and J.C. Minx (eds.)]. Cambridge University Press, Cambridge, United Kingdom and New York, NY, USA.

IPCC (2021) Climate Change 2021: The Physical Science Basis. Contribution of Working Group I to the

Sixth Assessment Report of the Intergovernmental Panel on Climate Change [Masson-Delmotte V, Zhai P, Pirani A, Connors SL, Péan C, Berger S, Caud N, Chen Y, Goldfarb L, Gomis MI, Huang M, Leitzell K, Lonnoy E, Matthews JBR, Maycock TK, Waterfield T, Yelekçi O, Yu R, Zhou B (eds.)]. Cambridge University Press. In Press

IPBES (2019): Global assessment report on biodiversity and ecosystem services of the Intergovernmental Science-Policy Platform on Biodiversity and Ecosystem Services. Brondizio ES, Settele J, Díaz S, Ngo HT (eds). IPBES secretariat, Bonn, Germany. 1148 pages. https://doi.org/10.5281/zenodo.3831673

Issa MA, Issa MA, Islam MS, Chudnovsky A (2003) Fractal dimension-a measure of fracture roughness and toughness of concrete. Eng Fract Mech 70:125–137. https://doi.org/10.1016/S0013-7944(02)00019-X

Jacob A, Peltz M, Hale S, et al (2021) Simulating permeability reduction by clay mineral nanopores in a tight sandstone by combining computer X-ray microtomography and focussed ion beam scanning electron microscopy imaging. Solid Earth 12:1–14. https://doi.org/10.5194/se-12-1-2021

James SC, Chrysikopoulos C V (1999) Transport of polydisperse colloid suspensions in a single fracture. Water Resour Res 35:707–718. https://doi.org/10.1029/1998WR900059

Johns RA, Steude JS, Castanier LM, Roberts P V. (1993) Nondestructive measurements of fracture aperture in crystalline rock cores using X ray computed tomography. J Geophys Res 98:1889–1900. https://doi.org/10.1029/92JB02298

Johnson CA, Piatak NM, Miller MM (2017) Barite (Barium). Chapter D in: Critical Mineral Resources of the United States—Economic and Environmental Geology and Prospects for Future Supply: U.S. Geological Survey Professional Paper 1802, Schulz KJ, DeYoung JH Jr., Seal II RR, Bradley DC (eds), USGS Publications Warehouse, Reston, VA, pp D1 – D18. https://doi.org/10.3133/pp1802

Jovanović Z, Khan F, Enzmann F, Kersten M (2013) Simultaneous segmentation and beam-hardening correction in computed microtomography of rock cores. Comput Geosci 56:142–150. https://doi.org/10.1016/j.cageo.2013.03.015

Ju Y, Dong J, Gao F, Wang J (2019) Evaluation of water permeability of rough fractures based on a self-affine fractal model and optimized segmentation algorithm. Adv Water Resour 129:99–111. https://doi.org/10.1016/j.advwatres.2019.05.007

Khan F (2014) Imaging and modeling of pore scale processes in porous media using X-ray computed tomography and lattice Boltzmann solver. Dissertation, Johannes-Gutenberg Universität Mainz: Mainz, Germany. http://doi.org/10.25358/openscience-4270

Kalare K, Bajpai M, Sarkar S, Munshi P (2021) Deep neural network for beam hardening artifacts removal in image reconstruction. Appl Intell. https://doi.org/10.1007/s10489-021-02604-y

Kamila Z, Kaya E, Zarrouk SJ (2021) Reinjection in geothermal fields: An updated worldwide review 2020. Geothermics 89:101970. https://doi.org/10.1016/j.geothermics.2020.101970

Kapur JN, Sahoo PK, Wong AKC (1985) A new method for gray-level picture thresholding using the entropy of the histogram. Comput Vision, Graph Image Process 29:273–285. https://doi.org/10.1016/0734-189X(85)90125-2

Karimpouli S, Tahmasebi P (2019) Segmentation of digital rock images using deep convolutional autoencoder networks. Comput Geosci 126:142–150. https://doi.org/10.1016/j.cageo.2019.02.003

Karimpouli S, Tahmasebi P, Saenger EH (2019) Coal Cleat/Fracture Segmentation Using Convolutional Neural Networks. Nat Resour Res. https://doi.org/10.1007/s11053-019-09536-y

Kato M, Takahashi M, Kawasaki S, et al (2013) Evaluation of porosity and its variation in porous materials using microfocus x-ray computed tomography considering the partial volume effect. Mater Trans 54:1678–1685. https://doi.org/10.2320/matertrans.M-M2013813

Kestin J, Sokolov M, Wakeham WA (1978) Viscosity of liquid water in the range −8 °C to 150 °C. J Phys Chem Ref Data 7:941–948. https://doi.org/10.1063/1.555581

Ketcham RA (2006) New algorithms for ring artifact removal. Dev X-Ray Tomogr V 6318:63180O. https://doi.org/10.1117/12.680939

Ketcham RA, Carlson WD (2001) Acquisition, optimization and interpretation of X-ray computed tomographic imagery: applications to the geosciences. Comput Geosci 27:381–400. https://doi.org/10.1016/S0098-3004(00)00116-3

Ketcham RA, Hanna RD (2014) Beam hardening correction for X-ray computed tomography of heterogeneous natural materials. Comput Geosci 67:49–61. https://doi.org/10.1016/j.cageo.2014.03.003

Ketcham RA, Slottke DT, Sharp JM (2010) Three-dimensional measurement of fractures in heterogeneous materials using high-resolution X-ray computed tomography. Geosphere 6:499–514. https://doi.org/10.1130/GES00552.1

Kioka A, Nakagawa M (2021) Theoretical and experimental perspectives in utilizing nanobubbles as inhibitors of corrosion and scale in geothermal power plant. Renew Sustain Energy Rev 149:111373. https://doi.org/10.1016/j.rser.2021.111373

Klaus T, Vollmer C, Werner K, Lehmann H, Müschen K (2010) Energieziel 2050: 100% Strom aus erneuerbaren Quellen. Published brochure of the Umweltbundesamt. https://www.umweltbundesamt.de/sites/default/files/medien/378/publikationen/energieziel_2050.pdf (accessed 22.02.2022)

Kling T, Huo D, Schwarz JO, et al (2016) Simulating stress-dependent fluid flow in a fractured core sample using real-time X-ray CT data. Solid Earth 7:1109–1124. https://doi.org/10.5194/se-7-1109-2016

Kling T, Schwarz JO, Wendler F, et al (2017) Fracture flow due to hydrothermally induced quartz growth. Adv Water Resour 107:93–107. https://doi.org/10.1016/j.advwatres.2017.06.011

Kluge C, Blöcher G, Hofmann H, et al (2021) The Stress-Memory Effect of Fracture Stiffness During Cyclic Loading in Low-Permeability Sandstone. J Geophys Res Solid Earth 126:. https://doi.org/10.1029/2020jb021469

Kluge C, Milsch H, Blöcher G (2017) Permeability of displaced fractures. Energy Procedia 125:88–97. https://doi.org/10.1016/j.egypro.2017.08.077

Knapp RB, Chiarappa ML, Durham WB (2000) An experimental exploration of the transport and capture of abiotic colloids in a single fracture. Water Resour Res 36:3139–3149. https://doi.org/10.1029/2000WR900145

Kong H, Akakin HC, Sarma SE (2013) A Generalized Laplacian of Gaussian Filter for Blob Detection and Its Applications. IEEE Trans Cybern 43:1719–1733. https://doi.org/10.1109/TSMCB.2012.2228639

Kong L, Ishutov S, Hasiuk F, Xu C (2021) 3D Printing for Experiments in Petrophysics, Rock Physics, and Rock Mechanics: A Review. SPE Reserv Eval Eng 24:721–732. https://doi.org/10.2118/206744-pa

Konzuk JS, Kueper BH (2004) Evaluation of cubic law based models describing single-phase flow through a rough-walled fracture. Water Resour Res 40:1–17.

https://doi.org/10.1029/2003WR002356

Kottwitz MO, Popov AA, Baumann TS, Kaus BJP (2020) The hydraulic efficiency of single fractures: Correcting the cubic law parameterization for self-affine surface roughness and fracture closure. Solid Earth 11:947–957. https://doi.org/10.5194/se-11-947-2020

Kovačević S, Radišić M, Mitrinović D, et al (2017) Tracer test and behavior of selected pharmaceuticals. Water Sci Technol Water Supply 17:1043–1052. https://doi.org/10.2166/ws.2016.202

Kovács D, Dabi G, Vásárhelyi B (2019) Image processing for fractal geometry-based discrete fracture network modeling input data: A methodological approach. Cent Eur Geol 62:1–14. https://doi.org/10.1556/24.61.2018.09

Krygier MC, LaBonte T, Martinez C, et al (2021) Quantifying the unknown impact of segmentation uncertainty on image-based simulations. Nat Commun 12:5414. https://doi.org/10.1038/s41467-021-25493-8

Kulenkampff J, Gründig M, Richter M, Enzmann F (2008) Evaluation of positron-emission-tomography for visualisation of migration processes in geomaterials. Phys Chem Earth 33:937–942. https://doi.org/10.1016/j.pce.2008.05.005

Kurale NG, Vaidya MV (2017) Retinal Hemorrhage Detection Using Splat Segmentation of Retinal Fundus Images. In: 2017 International Conference on Computing, Communication, Control and Automation (ICCUBEA). IEEE, pp 1–6

Kyle JR, Ketcham RA (2015) Application of high resolution X-ray computed tomography to mineral deposit origin, evaluation, and processing. Ore Geol Rev 65:821–839. https://doi.org/10.1016/j.oregeorev.2014.09.034

Lanczos C (1950) An Iteration Method for the Solution of the Eigenvalue Problem of Linear Differential and Integral Operators. Press Office, United States Governm.

Lang PS, Paluszny A, Nejati M, Zimmerman RW (2018) Relationship Between the Orientation of Maximum Permeability and Intermediate Principal Stress in Fractured Rocks. Water Resour Res 54:8734–8755. https://doi.org/10.1029/2018WR023189

Lead JR, Wilkinson KJ (2007) Environmental Colloids and Particles: Current Knowledge and Future Developments. Chapter 1 in: Environmental Colloids and Particles: Behaviour, Separation and Characterisation. Series on Analytical and Physical Chemistry of Environmental Systems.

Wilkinson KJ & Lead JR (eds), John Wiley & Sons, Hoboken, New Jersey, pp. 1 – 15.

Ledésert BA, Hébert RL, Mouchot J, et al (2021) Scaling in a Geothermal Heat Exchanger at Soultz-Sous-Forêts (Upper Rhine Graben, France): A XRD and SEM-EDS Characterization of Sulfide Precipitates. Geosciences 11:271. https://doi.org/10.3390/geosciences11070271

Lee D, Karadimitriou N, Ruf M, Steeb H (2022) Detecting micro fractures: a comprehensive comparison of conventional and machine-learning-based segmentation methods. Solid Earth 13:1475–1494. https://doi.org/10.5194/se-13-1475-2022

Lenchenkov N, Glasbergen G, van Kruijsdijk C (2018) Flow of a Cross-Linking Polymer in Porous Media. Transp Porous Media 124:943–963. https://doi.org/10.1007/s11242-018-1105-3

Leu L, Berg S, Enzmann F, et al (2014) Fast X-ray Micro-Tomography of Multiphase Flow in Berea Sandstone: A Sensitivity Study on Image Processing. Transp Porous Media 105:451–469. https://doi.org/10.1007/s11242-014-0378-4

Li B, Jiang Y, Koyama T, et al (2008a) Experimental study of the hydro-mechanical behavior of rock joints using a parallel-plate model containing contact areas and artificial fractures. Int J Rock Mech Min Sci 45:362–375. https://doi.org/10.1016/j.ijrmms.2007.06.004

Li K, Bian H, Liu C, et al (2015) Comparison of geothermal with solar and wind power generation systems. Renew Sustain Energy Rev 42:1464–1474. https://doi.org/10.1016/j.rser.2014.10.049

Li X, Huang H, Meakin P (2008b) Level set simulation of coupled advection-diffusion and pore structure evolution due to mineral precipitation in porous media. Water Resour Res 44:1–17. https://doi.org/10.1029/2007WR006742

Li Y, He X, Zhu W, et al (2022) Digital Rock Reconstruction Using Wasserstein GANs with Gradient Penalty. In: Day 1 Mon, February 21, 2022. IPTC

Li Y, Zhang Y (2015) Quantitative estimation of joint roughness coefficient using statistical parameters. Int J Rock Mech Min Sci 77:27–35. https://doi.org/10.1016/j.ijrmms.2015.03.016

Liang Y (2016) Rock fracture skeleton tracing by image processing and quantitative analysis by geometry features. J Geophys Eng 13:273–284. https://doi.org/10.1088/1742-2132/13/3/273

Linden S, Wiegmann A, Hagen H (2015) The LIR space partitioning system applied to the Stokes equations. Graph Models 82:58–66. https://doi.org/10.1016/j.gmod.2015.06.003

Liu R, He M, Huang N, et al (2020) Three-dimensional double-rough-walled modeling of fluid flow through self-affine shear fractures. J Rock Mech Geotech Eng 12:41–49. https://doi.org/10.1016/j.jrmge.2019.09.002

Lough MF, Lee SH, Kamath J (1998) An Efficient Boundary Integral Formulation for Flow Through Fractured Porous Media. J Comput Phys 143:462–483. https://doi.org/10.1006/jcph.1998.5858

Lu AYT, Shi W, Wang J, et al (2019) The Mechanism of Barium Sulfate Deposition Inhibition and the Prediction of Inhibitor Dosage. J Chem Eng Data 64:4968–4976. https://doi.org/10.1021/acs.jced.9b00799

Lund JW, Toth AN (2021) Direct utilization of geothermal energy 2020 worldwide review. Geothermics 90:1–39. https://doi.org/10.1016/j.geothermics.2020.101915

MacQueen J (1967) Some Methods for Classification and Analysis of Multivariate Observations. Proceedings of the 5th Berkeley Symposium on Mathematical Statistics and Probability, 1:281-297.

Magsipoc E, Zhao Q, Grasselli G (2020) 2D and 3D Roughness Characterization. Rock Mech Rock Eng 53:1495–1519. https://doi.org/10.1007/s00603-019-01977-4

Mao Q, Zhao S, Tong D, et al (2021) Hessian-MRLoG: Hessian information and multi-scale reverse LoG filter for pulmonary nodule detection. Comput Biol Med 131:104272. https://doi.org/10.1016/j.compbiomed.2021.104272

Martínez L (2018) Measuring the conductivity of very dilute electrolyte solutions, drop by drop. Quim Nova 41:814–817. https://doi.org/10.21577/0100-4042.20170216

Matsuki K, Lee JJ, Sakaguchi K, Hayashi K (1999) Size effect in flow conductance of a closed small-scale hydraulic fracture in granite. Geotherm Sci Technol 6:113–138

Milsch H, Hofmann H, Blöcher G (2016) An experimental and numerical evaluation of continuous fracture permeability measurements during effective pressure cycles. Int J Rock Mech Min Sci 89:109–115. https://doi.org/10.1016/j.ijrmms.2016.09.002

Mofakham AA, Stadelman M, Ahmadi G, et al (2018) Computational Modeling of Hydraulic Properties of a Sheared Single Rock Fracture. Transp Porous Media 124:1–30. https://doi.org/10.1007/s11242-018-1030-5

Molnar IL, Johnson WP, Gerhard JI, et al (2015) Predicting colloid transport through saturated porous media: A critical review. Water Resour Res 51:6804–6845. https://doi.org/10.1002/2015WR017318

Molnar IL, Pensini E, Asad MA, et al (2019) Colloid Transport in Porous Media: A Review of Classical Mechanisms and Emerging Topics. Transp Porous Media 130:129–156. https://doi.org/10.1007/s11242-019-01270-6

Mortensen AP (2001) Preferential flow phenomena in partially- saturated porous media. Ph.D. Thesis, Technical University of Denmark, Lyngby, Denmark.

Mosser L, Dubrule O, Blunt MJ (2017) Reconstruction of three-dimensional porous media using generative adversarial neural networks. Phys Rev E 96:. https://doi.org/10.1103/PhysRevE.96.043309

Mostaghimi P, Armstrong RT, Gerami A, et al (2017) Cleat-scale characterisation of coal: An overview. J Nat Gas Sci Eng 39:143–160. https://doi.org/10.1016/j.jngse.2017.01.025

Moya D, Aldás C, Kaparaju P (2018) Geothermal energy: Power plant technology and direct heat applications. Renew Sustain Energy Rev 94:889–901. https://doi.org/10.1016/j.rser.2018.06.047

Myers NO (1962) Characterization of surface roughness. Wear 5:182–189. https://doi.org/10.1016/0043-1648(62)90002-9

Neuzil CE, Tracy J V (1981) Flow through fractures. Water Resour Res 17:191–199. https://doi.org/10.1029/WR017i001p00191

Nitschke F, Scheiber J, Kramar U, Neumann T (2014) Formation of alternating layered Ba-Sr-sulfate and Pb-sulfide scaling in the geothermal plant of Soultz-sous-Forêts. Neues Jahrb für Mineral - Abhandlungen J Mineral Geoche 191:145–156. https://doi.org/10.1127/0077-7757/2014/0253

Niu Y, Mostaghimi P, Shabaninejad M, et al (2020) Digital Rock Segmentation for Petrophysical Analysis With Reduced User Bias Using Convolutional Neural Networks. Water Resour Res 56:0–2. https://doi.org/10.1029/2019WR026597

Njuguna J, Ansari F, Sachse S, et al. (2021): Nanomaterials, nanofillers, and nanocomposites: types and properties. In: Njuguna J, Pielichowski K, Zhu H (Eds.), Health and Environmental Safety of Nanomaterials - Polymer Nancomposites and Other Materials Containing Nanoparticles, 2nd ed., Woodhead Publishing, pp. 3 – 37.

Noiriel C, Madé B, Gouze P (2007) Impact of coating development on the hydraulic and transport properties in argillaceous limestone fracture. Water Resour Res 43:1–16. https://doi.org/10.1029/2006WR005379

Noiriel C, Renard F (2022) Four-dimensional X-ray micro-tomography imaging of dynamic processes in geosciences. Comptes Rendus Géoscience 354:255–280. https://doi.org/10.5802/crgeos.137

Noiriel C, Soulaine C (2021) Pore-Scale Imaging and Modelling of Reactive Flow in Evolving Porous Media: Tracking the Dynamics of the Fluid–Rock Interface. Transp Porous Media 140:181–213. https://doi.org/10.1007/s11242-021-01613-2

Oh W, Lindquist B (1999) Image thresholding by indicator kriging. IEEE Trans Pattern Anal Mach Intell 21:590–602. https://doi.org/10.1109/34.777370

Olsson R, Barton N (2001) An improved model for hydromechanical coupling during shearing of rock joints. Int J Rock Mech Min Sci 38:317–329. https://doi.org/10.1016/S1365-1609(00)00079-4

Orhan K, Vasconcelos KDF, Gaêta-Araujo H (2020) Artifacts in Micro-CT. In: Orhan K (ed) Micro-computed Tomography (micro-CT) in Medicine and Engineering, Springer Nature Switzerland AG 2020, Cham, pp 35 – 48. https://doi.org/10.1007/978-3-030-16641-0_4

Orywall P, Drüppel K, Kuhn D, et al (2017) Flow-through experiments on the interaction of sandstone with Ba-rich fluids at geothermal conditions. Geotherm Energy 5:20. https://doi.org/10.1186/s40517-017-0079-7

Oswald JG, Ibaraki M (2001) Migration of colloids in discretely fractured porous media: effect of colloidal matrix diffusion. J Contam Hydrol 52:213–244. https://doi.org/10.1016/S0169-7722(01)00159-0

Othman F, Yu M, Kamali F, Hussain F (2018) Fines migration during supercritical CO2 injection in sandstone. J Nat Gas Sci Eng 56:344–357. https://doi.org/10.1016/j.jngse.2018.06.001

Otsu N (1979) A Threshold Selection Method from Gray-Level Histograms. IEEE Trans Syst Man Cybern 9:62–66. https://doi.org/10.1109/TSMC.1979.4310076

Park HS, Chung YE, Seo JK (2015) Computed tomographic beam-hardening artefacts: mathematical characterization and analysis. Philos Trans R Soc A Math Phys Eng Sci 373:20140388. https://doi.org/10.1098/rsta.2014.0388

Phan J, Ruspini LC, Lindseth F (2021) Automatic segmentation tool for 3D digital rocks by deep learning. Sci Rep 11:1–16. https://doi.org/10.1038/s41598-021-98697-z

Ponomarev AA, Kadyrov MA, Tugushev OA, et al (2022) Digital core reconstruction research: challenges and prospects. Geol Ecol Landscapes 00:1–8. https://doi.org/10.1080/24749508.2022.2086201

Pot V, Zhong X, Baveye PC (2020) Effect of resolution, reconstruction settings, and segmentation methods on the numerical calculation of saturated soil hydraulic conductivity from 3D computed tomography images. Geoderma 362:114089. https://doi.org/10.1016/j.geoderma.2019.114089

Qi L, Cölfen H, Antonietti M (2000) Control of barite morphology by double-hydrophilic block copolymers. Chem Mater 12:2392–2403. https://doi.org/10.1021/cm0010405

Quan X, Fu Z, Yuan L, et al (2017) Capacitive deionization of NaCl solutions with ambient pressure dried carbon aerogel microsphere electrodes. RSC Adv 7:35875–35882. https://doi.org/10.1039/c7ra05226j

Rabbani A, Babaei M, Shams R, et al (2020) DeePore: A deep learning workflow for rapid and comprehensive characterization of porous materials. Adv Water Resour 146:103787. https://doi.org/10.1016/j.advwatres.2020.103787

Rabbani A, Fernando AM, Shams R, et al (2021) Review of Data Science Trends and Issues in Porous Media Research With a Focus on Image-Based Techniques. Water Resour Res 57:1–44. https://doi.org/10.1029/2020WR029472

Ramandi HL, Irtza S, Sirojan T, et al (2022) FracDetect: A novel algorithm for 3D fracture detection in digital fractured rocks. J Hydrol 127482. https://doi.org/10.1016/j.jhydrol.2022.127482

Ramandi HL, Mostaghimi P, Armstrong RT (2017) Digital rock analysis for accurate prediction of fractured media permeability. J Hydrol 554:817–826. https://doi.org/10.1016/j.jhydrol.2016.08.029

Raven KG, Gale JE (1985) Water flow in a natural rock fracture as a function of stress and sample size. Int J Rock Mech Min Sci Geomech Abstr 22:251–261. https://doi.org/10.1016/0148-9062(85)92952-3

Regenspurg S, Feldbusch E, Byrne J, et al (2015) Mineral precipitation during production of geothermal fluid from a Permian Rotliegend reservoir. Geothermics 54:122–135. https://doi.org/10.1016/j.geothermics.2015.01.003

Reinhardt M, Jacob A, Sadeghnejad S, et al (2022) Benchmarking conventional and machine learning segmentation techniques for digital rock physics analysis of fractured rocks. Environ Earth Sci

81:71. https://doi.org/10.1007/s12665-021-10133-7

Renard F, Voisin C, Marsan D, Schmittbuhl J (2006) High resolution 3D laser scanner measurements of a strike-slip fault quantify its morphological anisotropy at all scales. Geophys Res Lett 33:L04305. https://doi.org/10.1029/2005GL025038

Renshaw CE (1995) On the relationship between mechanical and hydraulic apertures in rough-walled fractures. J Geophys Res Solid Earth 100:24629–24636. https://doi.org/10.1029/95JB02159

Renshaw CE, Dadakis JS, Brown SR (2000) Measuring fracture apertures: A comparison of methods. Geophys Res Lett 27:289–292. https://doi.org/10.1029/1999GL008384

Rief S, Latz A, Wiegmann A (2006) Computer simulation of air filtration including electric surface charges in three-dimensional fibrous micro structures. Filtration 6, 169-172

Rolle M, Hochstetler D, Chiogna G, et al (2012) Experimental Investigation and Pore-Scale Modeling Interpretation of Compound-Specific Transverse Dispersion in Porous Media. Transp Porous Media 93:347–362. https://doi.org/10.1007/s11242-012-9953-8

Ronneberger O, Fischer P, Brox T (2015) U-Net: Convolutional Networks for Biomedical Image Segmentation. In: Navab N, Hornegger J, Wells WM, Frangi AF (eds) Medical Image Computing and Computer-Assisted Intervention -- MICCAI 2015. Springer International Publishing, Cham, pp 234–241

Roux S, Plouraboué F, Hulin JP (1998) Tracer Dispersion in Rough Open Cracks. Transp Porous Media 32:97–116. https://doi.org/10.1023/A:1006553902753

Ruiz-Agudo C, Putnis C V., Ruiz-Agudo E, Putnis A (2015a) The influence of pH on barite nucleation and growth. Chem Geol 391:7–18. https://doi.org/10.1016/j.chemgeo.2014.10.023

Ruiz-Agudo C, Ruiz-Agudo E, Putnis C V., Putnis A (2015b) Mechanistic Principles of Barite Formation: From Nanoparticles to Micron-Sized Crystals. Cryst Growth Des 15:3724–3733. https://doi.org/10.1021/acs.cgd.5b00315

Ryan JN, Elimelech M (1996) Colloid mobilization and transport in groundwater. Colloids Surfaces A Physicochem Eng Asp 107:1–56. https://doi.org/10.1016/0927-7757(95)03384-X

Saadat A, Frick S, Kranz S, Regenspurg S (2010) Energetic Use of EGS Reservoirs. In: Huenges E (ed) Geothermal Energy Systems: Exploration, Development and Utilization, Wiley-VCH, Weinheim, pp 303 - 372

Sadeghnejad S, Enzmann F, Kersten M (2021) Digital rock physics, chemistry, and biology: challenges and prospects of pore-scale modelling approaches. Appl Geochemistry 131:105028. https://doi.org/10.1016/j.apgeochem.2021.105028

Sadeghnejad S, Enzmann F, Kersten M (2022) Numerical Simulation of Particle Retention Mechanisms at the Sub-Pore Scale. Transp Porous Media. https://doi.org/10.1007/s11242-022-01843-y

Sanders JS, Fabian AC, Taylor GB, et al (2016) A very deep Chandra view of metals, sloshing and feedback in the Centaurus cluster of galaxies. Mon Not R Astron Soc 457:82–109. https://doi.org/10.1093/mnras/stv2972

Santos JE, Xu D, Jo H, et al (2020) PoreFlow-Net: A 3D convolutional neural network to predict fluid flow through porous media. Adv Water Resour 138:. https://doi.org/10.1016/j.advwatres.2020.103539

Saxena N, Hofmann R, Alpak FO, et al (2017a) References and benchmarks for pore-scale flow simulated using micro-CT images of porous media and digital rocks. Adv Water Resour 109:211–235. https://doi.org/10.1016/j.advwatres.2017.09.007

Saxena N, Hofmann R, Alpak FO, et al (2017b) Effect of image segmentation & voxel size on micro-CT computed effective transport & elastic properties. Mar Pet Geol 86:972–990. https://doi.org/10.1016/j.marpetgeo.2017.07.004

Saxena N, Hows A, Hofmann R, et al (2019) Rock properties from micro-CT images: Digital rock transforms for resolution, pore volume, and field of view. Adv Water Resour 134:103419. https://doi.org/10.1016/j.advwatres.2019.103419

Saxena N, Hows A, Hofmann R, et al (2018) Imaging and computational considerations for image computed permeability: Operating envelope of Digital Rock Physics. Adv Water Resour 116:127–144. https://doi.org/10.1016/j.advwatres.2018.04.001

Scanziani A, Singh K, Bultreys T, et al (2018) In situ characterization of immiscible three-phase flow at the pore scale for a water-wet carbonate rock. Adv Water Resour 121:446–455. https://doi.org/10.1016/j.advwatres.2018.09.010

Scarfe WC, Farman AG (2008) What is Cone-Beam CT and How Does it Work? Dent Clin North Am 52:707–730. https://doi.org/10.1016/j.cden.2008.05.005

Schäffer R, Götz E, Schlatter N, et al (2022) Fluid–Rock Interactions in Geothermal Reservoirs, Germany:

Thermal Autoclave Experiments Using Sandstones and Natural Hydrothermal Brines. Aquat Geochemistry. https://doi.org/10.1007/s10498-022-09404-x

Scheiber J, Nitschke F, Seibt A, Genter A (2012) Geochemical and mineralogical monitoring of the geothermal power plant in Soultz-Sous-Forêts (France). Thirty-Seventh Work Geotherm Reserv Eng Stanford Univ 10

Schlüter S, Sheppard A, Brown K, Wildenschild D (2014) Image processing of multiphase images obtained via X-ray microtomography: A review. Water Resour Res 50:3615–3639. https://doi.org/10.1002/2014WR015256

Schreiber S, Lapanje A, Ramsak P, Breembroek G. (2016) Operational issues in geothermal energy in Europe: status and overview. Reykjavik (Iceland): Geothermal ERA NET. http://www.geothermaleranet.is/media/publications/Geothermal-ERA-NET-JA-Report-OpERA_Publication.pdf (accessed 15.02.2022).

Schwartz LM, Martys N, Bentz DP, et al (1993) Cross-property relations and permeability estimation in model porous media. Phys Rev E 48:4584–4591. https://doi.org/10.1103/PhysRevE.48.4584

Schwarz JO, Enzmann F (2013) Simulation of Fluid Flow on Fractures and Implications for Reactive Transport Simulations. Transp Porous Media 96:501–525. https://doi.org/10.1007/s11242-012-0103-0

Sell K, Enzmann F, Kersten M, Spangenberg E (2013) Microtomographic quantification of hydraulic clay mineral displacement effects during a CO2 sequestration experiment with saline aquifer sandstone. Environ Sci Technol 47:198–204. https://doi.org/10.1021/es3013358

Shams R, Masihi M, Boozarjomehry RB, Blunt MJ (2020) Coupled generative adversarial and auto-encoder neural networks to reconstruct three-dimensional multi-scale porous media. J Pet Sci Eng 186:106794. https://doi.org/10.1016/j.petrol.2019.106794

Shelhamer E, Long J, Darrell T (2017) Fully Convolutional Networks for Semantic Segmentation. IEEE Trans Pattern Anal Mach Intell 39:640–651. https://doi.org/10.1109/TPAMI.2016.2572683

Shen C, Wang F, Li B, et al (2012) Application of DLVO Energy Map To Evaluate Interactions between Spherical Colloids and Rough Surfaces. Langmuir 28:14681–14692. https://doi.org/10.1021/la303163c

Shen N, Li X, Zhang Q, Wang L (2021) Comparison of shear-induced gas transmissivity of tensile fractures in sandstone and shale under varying effective normal stresses. J Nat Gas Sci Eng 95:104218. https://doi.org/10.1016/j.jngse.2021.104218

Shortall R, Davidsdottir B, Axelsson G (2015) Geothermal energy for sustainable development: A review of sustainability impacts and assessment frameworks. Renew Sustain Energy Rev 44:391–406. https://doi.org/10.1016/j.rser.2014.12.020

Shou YD, Zhao Z, Zhou XP (2020) Sensitivity analysis of segmentation techniques and voxel resolution on rock physical properties by X-ray imaging. J Struct Geol 133:103978. https://doi.org/10.1016/j.jsg.2020.103978

Siam M, Gamal M, Abdel-Razek M, et al (2018) RTSeg: Real-Time Semantic Segmentation Comparative Study. Proc - Int Conf Image Process ICIP 1603–1607. https://doi.org/10.1109/ICIP.2018.8451495

Simonyan K, Zisserman A (2015) Very deep convolutional networks for large-scale image recognition. 3rd Int Conf Learn Represent ICLR 2015 - Conf Track Proc 1–13

Song SB, Liu JF, Ni HY, et al (2020) A new automatic thresholding algorithm for unimodal gray-level distribution images by using the gray gradient information. J Pet Sci Eng 190:107074. https://doi.org/10.1016/j.petrol.2020.107074

Soret M, Bacharach SL, Buvat I (2007) Partial-Volume Effect in PET Tumor Imaging. J Nucl Med 48:932–945. https://doi.org/10.2967/jnumed.106.035774

Souley M, Lopez P, Boulon M, Thoraval A (2015) Experimental Hydromechanical Characterization and Numerical Modelling of a Fractured and Porous Sandstone. Rock Mech Rock Eng 48:1143–1161. https://doi.org/10.1007/s00603-014-0626-5

SPD, Bündnis 90/die Grünen und FDP (2021). Mehr Fortschritt wagen - Bündnis für Freiheit, Gerechtigkeit und Nachhaltigkeit. Koalitionsvertrag zwischen SPD, Bündnis 90/die Grünen und FDP für die 20. Legislaturperiode. https://www.spd.de/fileadmin/Dokumente/Koalitionsvertrag/Koalitionsvertrag_2021-2025.pdf (accessed 18.02.2022).

Spivakovskaya D, Heemink AW, Deleersnijder E (2007) The backward Îto method for the Lagrangian simulation of transport processes with large space variations of the diffusivity. Ocean Sci 3:525–535. https://doi.org/10.5194/os-3-525-2007

Stanchits S, Mayr S, Shapiro S, Dresen G (2011) Fracturing of porous rock induced by fluid injection.

Tectonophysics 503:129–145. https://doi.org/10.1016/j.tecto.2010.09.022

Steefel CI, Hu M (2022) Reactive Transport Modeling of Mineral Precipitation and Carbon Trapping in Discrete Fracture Networks. Water Resour Res 58:. https://doi.org/10.1029/2022WR032321

Stefansson V (2005) World geothermal assessment. In: Proceedings of the Conference World geothermal assessment. pp 24–29.

Stober I, Bucher K (2014) Geothermie. 2nd ed. Springer Verlag Berlin-Heidelberg. https://doi.org/10.1007/978-3-642-41763-4

Stoll M, Huber FM, Trumm M, et al (2019) Experimental and numerical investigations on the effect of fracture geometry and fracture aperture distribution on flow and solute transport in natural fractures. J Contam Hydrol 221:82–97. https://doi.org/10.1016/j.jconhyd.2018.11.008

Streit A, Becker J, Planas B (2020) GadGeo user guide, GeoDict 2021 from Math2Market GmbH, Germany, https://doi.org/10.30423/userguide.geodict2021-gadgeo (accessed 20.01.2022).

Suzuki A, Miyazawa M, Minto JM, et al (2021) Flow estimation solely from image data through persistent homology analysis. Sci Rep 11:17948. https://doi.org/10.1038/s41598-021-97222-6

Tan J, Rong G, Zhan H, et al (2020) An Innovative Method to Evaluate Hydraulic Conductivity of a Single Rock Fracture Based on Geometric Characteristics. Rock Mech Rock Eng 53:4767–4786. https://doi.org/10.1007/s00603-020-02196-y

Tsang YW (1992) Usage of “equivalent apertures” for rock fractures as derived from hydraulic and tracer tests. Water Resour Res 28:1451–1455. https:// doi. org/ 10. 1029/ 92WR0 0361

Tang K, Da Wang Y, McClure J, et al (2022) Generalizable Framework of Unpaired Domain Transfer and Deep Learning for the Processing of Real-Time Synchrotron-Based X-Ray Microcomputed Tomography Images of Complex Structures. Phys Rev Appl 17:034048. https://doi.org/10.1103/PhysRevApplied.17.034048

Tang Y, Yang F, Yuan S, Zhan C (2019) A MULTI-STAGE FRAMEWORK WITH CONTEXT INFORMATION FUSION STRUCTURE FOR SKIN LESION SEGMENTATION Guangdong Provincial Key Laboratory for Medical Image Processing , School of Biomedical. 2019 IEEE 16th Int Symp Biomed Imaging (ISBI 2019) 1407–1410

Taud H, Martinez-Angeles R, Parrot JF, Hernandez-Escobedo L (2005) Porosity estimation method by X-ray computed tomography. J Pet Sci Eng 47:209–217.

https://doi.org/10.1016/j.petrol.2005.03.009

Tawfik MS, Adishesha AS, Hsi Y, et al (2022) Comparative Study of Traditional and Deep-Learning Denoising Approaches for Image-Based Petrophysical Characterization of Porous Media. Front Water 3:. https://doi.org/10.3389/frwa.2021.800369

Tranter M, De Lucia M, Kühn M (2021a) Barite Scaling Potential Modelled for Fractured-Porous Geothermal Reservoirs. Minerals 11:1198. https://doi.org/10.3390/min11111198

Tranter M, De Lucia M, Kühn M (2021b) Numerical investigation of barite scaling kinetics in fractures. Geothermics 91:102027. https://doi.org/10.1016/j.geothermics.2020.102027

Tranter M, De Lucia M, Wolfgramm M, Kühn M (2020) Barite scale formation and injectivity loss models for geothermal systems. Water (Switzerland) 12:1-. https://doi.org/10.3390/w12113078

Tsang YW, Witherspoon PA (1981) Hydromechanical behavior of a deformable rock fracture subject to normal stress. J Geophys Res Solid Earth 86:9287–9298. https://doi.org/10.1029/JB086iB10p09287

United Nations (2015) General Assembly Resolution A/RES/70/1. Transforming Our World, the 2030 Agenda for Sustainable Development. https://sdgs.un.org/2030agenda (accessed 17.02.2022).

UNFCCC (2015) Decision 1/CP.21: Adoption of the Paris Agreement. FCCC/CP.2015/10/Add.1. https://unfccc.int/sites/default/files/resource/docs/2015/cop21/eng/10a01.pdf (accessed 17.02.2022).

Valsala R, Govindarajan SK (2022) Numerical modeling of colloid-assisted BTEX transport in a saturated fractured aquifer. Environ Earth Sci 81:1–15. https://doi.org/10.1007/s12665-021-10161-3

Van Beers F, Lindström A, Okafor E, Wiering MA (2019) Deep neural networks with intersection over union loss for binary image segmentation. ICPRAM 2019 - Proc 8th Int Conf Pattern Recognit Appl Methods 438–445. https://doi.org/10.5220/0007347504380445

Van Geet M, Swennen R (2001) Quantitative 3D-fracture analysis by means of microfocus X-Ray Computer Tomography (µCT): An example from coal. Geophys Res Lett 28:3333–3336. https://doi.org/10.1029/2001GL013247

Varfolomeev I, Yakimchuk I, Safonov I (2019) An application of deep neural networks for segmentation of microtomographic images of rock samples. Computers 8:1–21. https://doi.org/10.3390/computers8040072

Vásárhelyi L, Kónya Z, Kukovecz, Vajtai R (2020) Microcomputed tomography–based characterization of advanced materials: a review. Mater Today Adv 8:1–13. https://doi.org/10.1016/j.mtadv.2020.100084

Veerman J (2020) The effect of the NaCl bulk concentration on the resistance of ion exchange membranes-measuring and modeling. Energies 13:. https://doi.org/10.3390/en13081946

Vilks P, Bachinski DB (1996) Colloid and suspended particle migration experiments in a granite fracture. J Contam Hydrol 21:269–279. https://doi.org/10.1016/0169-7722(95)00053-4

Vilks P, Miller NH, Vorauer A (2008) Laboratory bentonite colloid migration experiments to support the Äspö Colloid Project. Phys Chem Earth, Parts A/B/C 33:1035–1041. https://doi.org/10.1016/j.pce.2008.05.013

Vinegar HJ, De Waal JA, Wellington SL (1991) CT studies of brittle failure in castlegate sandstone. Int J Rock Mech Min Sci Geomech Abstr 28:441–450. https://doi.org/10.1016/0148-9062(91)90082-W

Vogler D, Settgast RR, Annavarapu C, et al (2018) Experiments and Simulations of Fully Hydro-Mechanically Coupled Response of Rough Fractures Exposed to High-Pressure Fluid Injection. J Geophys Res Solid Earth 123:1186–1200. https://doi.org/10.1002/2017JB015057

Voorn M, Exner U, Rath A (2013) Multiscale Hessian fracture filtering for the enhancement and segmentation of narrow fractures in 3D image data. Comput Geosci 57:44–53. https://doi.org/10.1016/j.cageo.2013.03.006

Wallace JM, Hobbs PV (2006) Atmospheric Science: an introductory survey. 2nd ed. Elsevier Academic Press, Amsterdam.

Wang Y Da, Armstrong RT, Mostaghimi P (2020a) Boosting Resolution and Recovering Texture of 2D and 3D Micro-CT Images with Deep Learning. Water Resour Res 56:1–43. https://doi.org/10.1029/2019WR026052

Wang Y Da, Blunt MJ, Armstrong RT, Mostaghimi P (2021a) Deep learning in pore scale imaging and modeling. Earth-Science Rev 215:103555. https://doi.org/10.1016/j.earscirev.2021.103555

Wang Y Da, Shabaninejad M, Armstrong RT, Mostaghimi P (2021b) Deep neural networks for improving physical accuracy of 2D and 3D multi-mineral segmentation of rock micro-CT images. Appl Soft Comput 104:107185. https://doi.org/10.1016/j.asoc.2021.107185

Wang H, Dalton L, Fan M, et al (2022) Deep-learning-based workflow for boundary and small target segmentation in digital rock images using UNet++ and IK-EBM. J Pet Sci Eng 215:. https://doi.org/10.1016/j.petrol.2022.110596

Wang L, Cardenas MB (2015) An efficient quasi-3D particle tracking-based approach for transport through fractures with application to dynamic dispersion calculation. J Contam Hydrol 179:47–54. https://doi.org/10.1016/j.jconhyd.2015.05.007

Wang L, Cardenas MB, Slottke DT, et al (2015) Modification of the Local Cubic Law of fracture flow for weak inertia, tortuosity, and roughness. Water Resour Res 51:2064–2080. https://doi.org/10.1002/2014WR015815

Wang L, Cardenas MB, Zhou J, Ketcham RA (2020b) The Complexity of Nonlinear Flow and non-Fickian Transport in Fractures Driven by Three-Dimensional Recirculation Zones. J Geophys Res Solid Earth 125:1–14. https://doi.org/10.1029/2020JB020028

Wang S, Li W, Wang Y, et al (2012) An Improved Difference of Gaussian Filter in Face Recognition. J Multimed 7:387–393. https://doi.org/10.4304/jmm.7.6.429-433

Wang Y, Arns CH, Rahman SS, Arns J-Y (2018) Porous Structure Reconstruction Using Convolutional Neural Networks. Math Geosci 50:781–799. https://doi.org/10.1007/s11004-018-9743-0

Wang Z, Yang S, Tang Y (2020c) Mechanical behavior of different sedimentary rocks in the Brazilian test. Bull Eng Geol Environ 79:5415–5432. https://doi.org/10.1007/s10064-020-01906-8

Watanabe N, Ishibashi T, Ohsaki Y, et al (2011) X-ray CT based numerical analysis of fracture flow for core samples under various confining pressures. Eng Geol 123:338–346. https://doi.org/10.1016/j.enggeo.2011.09.010

Weber A, Streit A, Planas B (2020) AddiDict user guide, GeoDict 2021 from Math2Market GmbH, Germany, https://doi.org/10.30423/userguide.geodict2021-addidict (accessed 20.01.2022).

Wenning QC, Madonna C, Kurotori T, et al (2021) Chemo-Mechanical Coupling in Fractured Shale With Water and Hydrocarbon Flow. Geophys Res Lett 48:. https://doi.org/10.1029/2020GL091357

Wenning QC, Madonna C, Kurotori T, Pini R (2019) Spatial Mapping of Fracture Aperture Changes With Shear Displacement Using X-ray Computerized Tomography. J Geophys Res Solid Earth 124:7320–7340. https://doi.org/10.1029/2019JB017301

Widanagamage IH, Waldron AR, Glamoclija M (2018) Controls on barite crystal morphology during abiotic precipitation. Minerals 8:5–10. https://doi.org/10.3390/min8110480

Wildenschild D, Hopmans JW, Vaz CMP, et al (2002) Using X-ray computed tomography in hydrology: systems, resolutions, and limitations. (Special Issue: Non-invasive methods in hydrology). J Hydrol 267:285–297

Wildenschild D, Rivers ML, Porter ML, et al (2015) Using Synchrotron-Based X-Ray Microtomography and Functional Contrast Agents in Environmental Applications. 1–22. https://doi.org/10.2136/sssaspecpub61.c1

Withers PJ, Bouman C, Carmignato S, et al (2021) X-ray computed tomography. Nat Rev Methods Prim 1:18. https://doi.org/10.1038/s43586-021-00015-4

Witherspoon PA, Wang JSY, Iwai K, Gale JE (1980) Validity of Cubic Law for fluid flow in a deformable rock fracture. Water Resour Res 16:1016–1024. https://doi.org/10.1029/WR016i006p01016

Xie LZ, Gao C, Ren L, Li CB (2015) Numerical investigation of geometrical and hydraulic properties in a single rock fracture during shear displacement with the Navier–Stokes equations. Environ Earth Sci 73:7061–7074. https://doi.org/10.1007/s12665-015-4256-3

Xiao W, Xia C, Wei W, Bian Y (2013) Combined effect of tortuosity and surface roughness on estimation of flow rate through a single rough joint. J Geophys Eng 10:. https://doi.org/10.1088/1742-2132/10/4/045015

Xiong X, Li B, Jiang Y, et al (2011) Experimental and numerical study of the geometrical and hydraulic characteristics of a single rock fracture during shear. Int J Rock Mech Min Sci 48:1292–1302. https://doi.org/10.1016/j.ijrmms.2011.09.009

Yadava GK, Pal D, Hsieh J (2014) Reduction of metal artifacts: beam hardening and photon starvation effects. In: Whiting BR, Hoeschen C (eds) Medical Imaging 2014: Physics of Medical Imaging. p 90332V

Yang E, Kang DH, Yun TS (2020) Reliable estimation of hydraulic permeability from 3D X-ray CT images of porous rock. E3S Web Conf 205:1–5. https://doi.org/10.1051/e3sconf/202020508004

Yeo IW, de Freitas MH, Zimmerman RW (1998) Effect of shear displacement on the aperture and permeability of a rock fracture. Int J Rock Mech Min Sci 35:1051–1070. https://doi.org/10.1016/S0148-9062(98)00165-X

Yosri A, Dickson-Anderson S, Siam A, El-Dakhakhni W (2021) Analytical description of colloid behavior in single fractures under irreversible deposition. J Colloid Interface Sci 589:597–604. https://doi.org/10.1016/j.jcis.2020.12.089

Yu J, Liu S, Cheng B (2005) Effects of PSMA additive on morphology of barite particles. J Cryst Growth 275:572–579. https://doi.org/10.1016/j.jcrysgro.2004.12.033

Yuan Y, Chao M, Lo YC (2017) Automatic Skin Lesion Segmentation Using Deep Fully Convolutional Networks with Jaccard Distance. IEEE Trans Med Imaging 36:1876–1886. https://doi.org/10.1109/TMI.2017.2695227

Zambrano M, Hameed F, Anders K, et al (2019a) Implementation of Dynamic Neutron Radiography and Integrated X-Ray and Neutron Tomography in Porous Carbonate Reservoir Rocks. Front Earth Sci 7:. https://doi.org/10.3389/feart.2019.00329

Zambrano M, Pitts AD, Salama A, et al (2019b) Analysis of fracture roughness control on permeability using SFM and fluid flow simulations: Implications for carbonate reservoir characterization. Geofluids 2019:. https://doi.org/10.1155/2019/4132386

Zatz LM (1981) Basic principles of computed tomography scanning. In: Newton TH, Potts DG (Eds.), Technical Aspects of Computed Tomography. Mosby, St. Louis, pp. 3853–3876.

Zhang P, Lee Y Il, Zhang J (2019) A review of high-resolution X-ray computed tomography applied to petroleum geology and a case study. Micron 124:102702. https://doi.org/10.1016/j.micron.2019.102702

Zhang Q, Deng H, Dong Y, et al (2022) Investigation of Coupled Processes in Fractures and the Bordering Matrix via a Micro-Continuum Reactive Transport Model. Water Resour Res 58:1–18. https://doi.org/10.1029/2021WR030578

Zhang W, Tang X, Weisbrod N, Guan Z (2012) A review of colloid transport in fractured rocks. J Mt Sci 9:770–787. https://doi.org/10.1007/s11629-012-2443-1

Zhang Z, Nemcik J, Qiao Q, Geng X (2015) A Model for Water Flow Through Rock Fractures Based on Friction Factor. Rock Mech Rock Eng 48:559–571. https://doi.org/10.1007/s00603-014-0562-4

Zhao H, Huang Y, Deng S, et al (2022) Research progress on scaling mechanism and anti-scaling technology of geothermal well system. J Dispers Sci Technol 0:1–14. https://doi.org/10.1080/01932691.2022.2033625

Zhao Z, Li B, Jiang Y (2014) Effects of Fracture Surface Roughness on Macroscopic Fluid Flow and Solute

Transport in Fracture Networks. Rock Mech Rock Eng 47:2279–2286. https://doi.org/10.1007/s00603-013-0497-1

Zheng Q, Zhang D (2022) Digital Rock Reconstruction with User-Defined Properties Using Conditional Generative Adversarial Networks. Transp Porous Media 144:255–281. https://doi.org/10.1007/s11242-021-01728-6

Zhou J-Q, Li C, Wang L, et al (2021) Effect of slippery boundary on solute transport in rough-walled rock fractures under different flow regimes. J Hydrol 598:126456. https://doi.org/10.1016/j.jhydrol.2021.126456

Zimmerman RW, Bodvarsson GS (1996) Hydraulic conductivity of rock fractures. Transp Porous Media 23:1–30. https://doi.org/10.1007/BF00145263

Zimmerman RW, Kumar S, Bodvarsson GS (1991) Lubrication theory analysis of the permeability of rough-walled fractures. Int J Rock Mech Min ScL Geomech 28:535–541. https://doi.org/10.1016/0148-9062(92)91194-a

Zimmerman RW, Yeo IW (2000) Fluid flow in rock fractures: From the navier-stokes equations to the cubic law. Geophys Monogr Ser 122:213–224. https://doi.org/10.1029/GM122p0213

Zimmermann G, Moeck I, Blöcher G (2010) Cyclic waterfrac stimulation to develop an Enhanced Geothermal System (EGS)—Conceptual design and experimental results. Geothermics 39:59–69. https://doi.org/10.1016/j.geothermics.2009.10.003

Zou L, Jing L, Cvetkovic V (2017) Modeling of Solute Transport in a 3D Rough-Walled Fracture–Matrix System. Transp Porous Media 116:1005–1029. https://doi.org/10.1007/s11242-016-0810-z

Zvikelsky O, Weisbrod N (2006) Impact of particle size on colloid transport in discrete fractures. Water Resour Res 42:1–12. https://doi.org/10.1029/2006WR004873