A Comparison of Two Numerical Models for Flow in Fractured Porous Media and the Impact of Fracture Intersection Cell Removal

Master of Science Thesis in Applied and Computational Mathematics

Ivar Stefansson

Department of Mathematics

University of Bergen

November, 2016

Acknowledgements

I want to most warmly thank my supervisors, Inga Berre and Jan M. Nordbotten, for all the help they have graciously given. Their kind guidance and enthusiasm have been most encouraging and I have much appreciated how their doors have always been open to me. I would also like to thank Wietse Boon, whose patient advice and cheerful discussions I have sincerely enjoyed.

My thanks also go to my friends and family for support and welcome distractions. I am also grateful to everyone who has asked questions and shown interest in the thesis and its topic.

Abstract

The simulation of flow and transport in fractured porous media is crucial to improve understanding and viability of geothermal energy. It is also challenging, due to complex geometries and large scale differences. In this thesis, we compare two numerical methods developed to meet these needs; a Finite Volume method and a Mixed Finite Element method. We analyse their behaviour based on a collection of test cases. In addition to studying the produced pressure solutions, we perform transport simulations to reveal local behaviour through accumulation.

We take particular interest in the intersections of fractures and the possibilities in removing the intersection cells from the spatial discretization, improving condition numbers and relaxing time step restrictions. We identify some dangers in eliminating the cells before the flow is solved, and instead propose a procedure to be applied between the flow simulation and the transport simulation.

Table of Contents

1. Motivation and Outline

Flow of fluids through fractured porous media is a key component in the modelling of a range of phenomena. Examples include groundwater flow, storage of gases such as CO_2 in the subsurface, contamination remediation and geothermal energy production. We will in the following concentrate on the geothermal application. As a renewable energy resource of vast potential, the hope is that its exploitation may contribute to the replacement of fossil energy sources.

To better understand and predict the processes taking place in geothermal reservoirs, physical models are designed based on the particular physics one wishes to cover. Choices have to be made regarding how many fluids one accounts for, to which extent their mixing is resolved etc. In some of the applications, including geothermal energy production, the temperature of the fluid and the porous medium is of central importance. In this case, the model has to include the energy of the system. Once the physical model is ready, it has to be formulated mathematically. As the resulting set of equations is not in general analytically solvable, one then applies a numerical model to obtain a solution.

Of course, we are interested in the quality of those simulations, both in terms of the ability to accurately solve the problem and the computational cost of doing so. The objective of this thesis is to evaluate two numerical schemes developed at the University of Bergen. We will place particular emphasis on the behaviour at fracture intersections and suggest a possible improvement in how these are handled.

The thesis is structured as follows:

We set out by presenting general theory of porous media and fractures in Chapter 2. We describe the assumptions made for modelling fluid flow and heat transfer. Towards the end of the chapter, we arrive at the set of partial differential equations to be solved by the numerical methods.

In Chapter 3, we describe the general process of discretizing a spatial domain and the particulars of how this is done for the two methods for solving the flow. Then we do the same for the transport discretization, including a section on temporal discretization. We introduce the new procedure for eliminating intersection cells after the flow simulation and before the transport simulation. Finally, we describe the implementation and the means by which the results are compared.

Then, in Chapter 4, we present and discuss a series of test cases. Starting from an idealized case where individual effects are readily identifiable, we progress in order of increasing complexity till we reach a case of realistic geometry.

The main findings and conclusions are summarized in Chapter 5.

In the attachment in Section 7 we include a comparison paper presenting results for the two aforementioned flow methods and four other. Some of the results of Section 4 are based on those presented in the paper.

2. Theory of Fractured Porous Media

This chapter gives a brief introduction to the theory of flow and heat transport in porous media and rock fractures. Starting from definitions and basic physics, we arrive at a set of equations which we will later solve using numerical models.

2.1 Porous Media and Representative Elementary Volumes

By the term "porous medium", we refer to a solid penetrated by an interconnected set of pores, allowing fluids to flow through the medium. In naturally occurring media such as subsurface rocks of a geothermal reservoir, we must expect the small scale pores to be heterogeneously distributed in space. To avoid accounting for each small-scale pore or heterogeneity explicitly, we assume we can average the properties of the rock over some spatial domain. For this approach to be sound, however, the property being averaged must satisfy certain conditions in the spatial region we want to average over. To describe this, we use the concept of the representative elementary volume, or the REV. This volume lies inside the length scale interval where the small scale (molecular) variations are smoothed out and the large scale variations, those we are interested in computing, are not yet relevant, see Figure 1. We must, in other words, assume that there is some intermediate length scale where the parameter in question does not change significantly. As pointed out in (Bear, 2010), the concept of a porous medium should only be used if a common REV can be defined throughout the domain. If not, the modelling approach based on averaged values of the parameters that is used is not valid.

Neglecting the unconnected pores, we can define the effective porosity, hereafter simply the porosity, within the REV as

$$\phi = \phi_E = \frac{V_{EP}}{V_T}, \tag{1}$$

where V_T and V_{EP} are the total volume and the volume of the interconnected pores of the REV, respectively. For a visualization of the concept, see Figure 1. A displacement on the molecular scale (left) might very well change the value of the parameter drastically; the value changes from 1 to 0 if we happen to move from inside a small scale pore to the rock. On the intermediate scale, these heterogeneities are averaged out, as the number of the small-scale pores is so high. On the other hand, we expect large scale fluctuations in porosity if the medium is heterogeneous, due to e.g. different geological layers, as indicated to the right.

Analogously, we define parameters of the fluid for the same volume. The dynamic viscosity μ, which describes the fluid's resistance to flow, and the density ρ both fluctuate on the molecular scale, but not on the intermediate one.

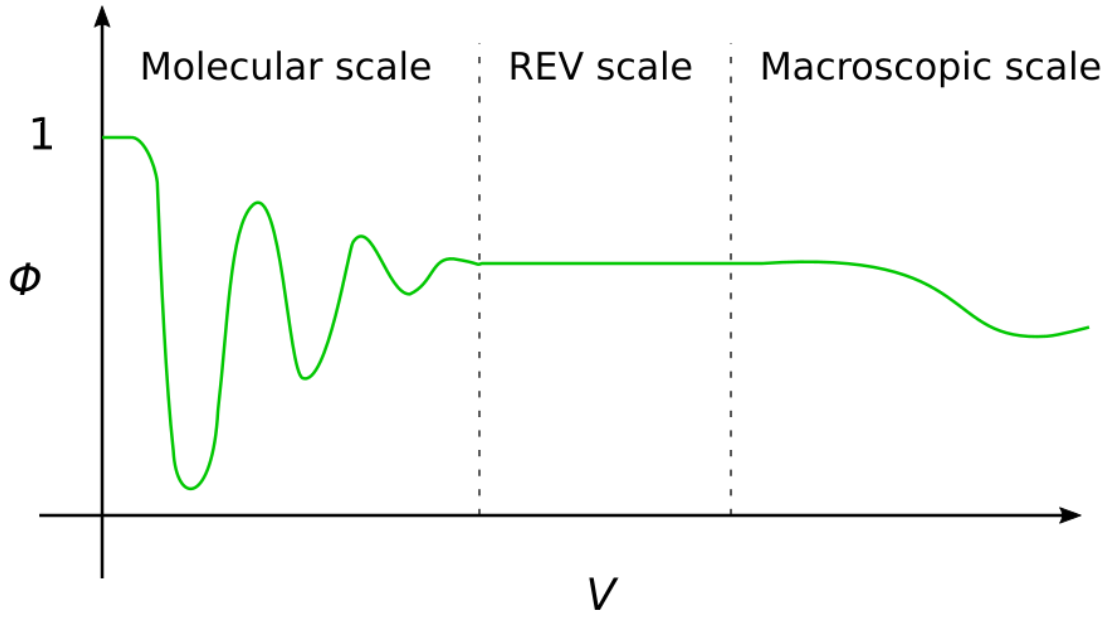


Figure 1. The porosity might fluctuate both below and above, but not on, the REV scale in a heterogeneous medium.

2.2 Darcy's Law

The flow of a given fluid through a porous medium depends on the properties of the medium. The relationship was described by Henry Darcy in 1856 after a series of experiments (Nield & Bejan, 2006). His findings can in modern notation be summarized in what is known as Darcy's law:

$$\boldsymbol{u} = -\boldsymbol{K}'\nabla h. \tag{2}$$

Here, $\boldsymbol{u}$ denotes the volumetric flux vector in units of volume per area per time, $\boldsymbol{K}'$ is the hydraulic conductivity tensor and h is the hydraulic head. The conductivity quantifies the material properties determining how easily a flow is induced. To account for anisotropic media, it is given the form of a $d \times d$ matrix, where d is thee spatial dimension of the model. The conductivity depends on the properties of the porous medium and fluid through the values of the intrinsic rock permeability $\boldsymbol{K}$ and the fluid's density and viscosity, respectively. Denoting the gravity constant by g, we have

$$\boldsymbol{K}' = \boldsymbol{K}\frac{\rho g}{\mu}. \tag{3}$$

Just as the parameters introduced in Section 2.1, the hydraulic conductivity and rock permeability are also defined on the REV scale.

The hydraulic head is a measure of the fluid pressure relative to some given datum and is given by

$$h = \frac{p}{\rho g} + z, \tag{4}$$

where p is the pressure, g is the gravitational acceleration and z the elevation above the same datum. It can be seen as an indicator of the potential energy of the fluid, and its importance for flow in porous media lies in the fact that the flow will be directed from regions of high hydraulic head to regions of lower values of h (Nordbotten & Celia, 2012).

There is one important assumption for Darcy's law which needs to be mentioned, also because we will use this assumption directly in Section 3.6: The flow must be relatively slow. A rule of thumb is that the Reynolds number, the ratio between inertial and viscous forces, should be below 1. This implies that all the fluid within each pore flows in parallel in the same direction without lateral mixing; it is laminar. This assumption is justified in most porous media flow scenarios, but there are exceptions where one should be cautious. One example is highly permeable fractures, where the flow may be orders of magnitude larger than in the rest of the domain, as discussed in Section 2.5.

2.3 Conservation of Mass

We will model the kinematics of the fluid using the principle of mass conservation, which states that the mass of a closed system remains constant. The mathematical formulation of this is that the time derivative of the integral of the mass over an arbitrary volume V in space equals zero

$$\frac{d}{dt}\int_V m\, dV = 0, \tag{5}$$

with m denoting specific mass, or mass per volume. Now, the systems we consider are not necessarily closed, but subject to external influences in the form of flow over the boundaries. We can also have internal production or extraction of mass, think of a well in a groundwater simulation. These internal terms are referred to as sources or sinks. The corresponding mass conservation equation must take both types of terms into account

$$\frac{d}{dt}\int_V m\, dV = -\int_A \boldsymbol{u}\cdot\boldsymbol{n} dA + \int_V s\, dV. \tag{6}$$

Here, $\boldsymbol{n}$ denotes the outward normal vector of the domain surface and s the sources. The negative sign of the first right hand side term ensures that a flux out of the domain ($\boldsymbol{u}$ and $\boldsymbol{n}$ in the same direction) corresponds to a decrease in the left hand side, which still represents the mass in the domain. When we take the scalar product of a flux with an oriented elementary area $\boldsymbol{n}dA$, we

obtain the specific or elementary discharge, and integrating over some area A, we find the total discharge, which tells us how much fluid passes over the area per unit time (Bear & Bachmat, 1991).

The surface integral term can be converted into a volume integral by applying the divergence theorem

$$\int_A \boldsymbol{u} \cdot \boldsymbol{n} dA = \int_V \nabla \cdot \boldsymbol{u} dV. \tag{7}$$

Further, the time derivative and integral may be interchanged for the left hand side using Leibniz' integral rule under the assumptions of sufficiently smooth integrand, and our mass conservation equation takes the form

$$\int_V \frac{\partial m}{\partial t} + \nabla \cdot (\boldsymbol{u}) - s dV = 0. \tag{8}$$

At this point, we can apply the fundamental lemma of the calculus of variations. It states that any function whose integral over an arbitrary volume is equal to zero is itself equal to zero. This gives the differential form of the conservation equation

$$\frac{\partial m}{\partial t} + \nabla \cdot \boldsymbol{u} - s = 0. \tag{9}$$

Some simplifications bring us to what is sometimes referred to as the groundwater equation. First, we notice that we can write the mass of a substance as a product of its density and volume. Given that we assume only one fluid to be present, and assuming that it has to take up the entire pore space, the volume equals the total volume times the porosity, or the amount of available pore space in the medium. Thus, the specific mass is

$$m = \rho \phi. \tag{10}$$

Throughout this thesis, we will assume both the porosity and the density to be constant, so the time derivative term of Equation 9 vanishes altogether. The assumption of constant porosity is valid in cases where the changes in the REV scale rock structure are negligible on the time scale of interest, and may be reasonable e.g. in the production period of a geothermal energy plant. As the density is dependent on pressure and temperature, the latter assumption requires that the temperature and pressure range of the domain is not too large. However, the density of fluids such as water is far less sensitive to pressure changes than that of gases, and so the assumption is not too restrictive in the current setting.

Applying Darcy's law for the flux $\boldsymbol{u}$ and inserting into the conservation equation yields

$$-\nabla \cdot (\boldsymbol{K}'\nabla h) - s = 0. \tag{11}$$

We will throughout this thesis neglect gravity effects, and thus the hydraulic head and pressure are equal up to a constant $\rho g \mu^{-1}$. Scaling the permeability tensor with the inverse of the viscosity leaves us with

$$-\nabla \cdot (\boldsymbol{K}\nabla p) = s, \tag{12}$$

which is the equation we will apply the flow schemes to. They will produce a pressure and flow field, the latter of which we will take for further analysis when we use them in heat transport simulations.

2.4 Conservation of Energy

By introducing a second conservation equation, namely for energy, one can also determine the distribution of the heat. In general, this leads to a coupled problem where the fluid movement depend on the heat distribution and the heat fluxes on the fluid. But decoupling is possible by making simplifying assumptions. As long as $\boldsymbol{K}$ does not depend on the temperature through ρ and μ, the two problems can be solved sequentially.

Similar to the case of mass conservation, the foundation for the heat equation is that the change in energy of any arbitrary system must equal the energy added or removed over the boundaries and through internal sources and sinks. Denoting the energy by E, energy flux by $\boldsymbol{J}$ and sources by f' and performing the exact same steps as for the mass conservation above, the equation takes the form

$$\frac{\partial E}{\partial t} + \nabla \cdot \boldsymbol{J} = f'. \tag{13}$$

In general, the energy equals the sum of the potential, kinetic and internal energy. For the purposes of this thesis, we will only include the last term, as it exceeds the other two in magnitude in most relevant applications as long as we consider incompressible fluids and no phase change. The internal energy is strongly associated with the fluid's heat and denoted by $U_{internal}$. In the integral form, we need its specific counterpart (per volume) u, defined by

$$U_{internal} = \int_V \rho u_{internal}\, dV. \tag{14}$$

Under the assumptions of slowly changing pressures within the REV, the change in internal

energy is a linear function of the change in temperature

$$du_{internal} = c_p dT, \tag{15}$$

with c_p being the heat capacity at constant pressure.

We will further assume that the energy flux consists of two terms; one diffusive and one advective part. This means we neglect the volume changing pressure work and is valid for fluids in most pressure regimes, but should not be done for gases (Class, 2007). The first is the diffusion of energy from regions of high energy concentration to regions of low concentration, whereas the last is the energy that is transported with the flow:

$$\boldsymbol{J} = \boldsymbol{J}_{diff} + \boldsymbol{J}_{adv}. \tag{16}$$

The diffusive transport is modelled very similarly to the flow-pressure relation in Darcy's law. By Fourier's law, it is given as

$$\boldsymbol{J}_{diff} = -\boldsymbol{\kappa}' \nabla T. \tag{17}$$

Here, T is the temperature and $\boldsymbol{\kappa}'$ is the heat conduction tensor, the heat spreading equivalent of the permeability tensor. In addition to the purely conductive heat spreading, the term accounts for the dispersive mixing due to sub-REV scale flow field heterogeneities (Bear & Bachmat, 1991).

Once the flux field $\boldsymbol{u}$ has been computed, the advective term accounting for the heat transported by the movement of the fluid is simply

$$\boldsymbol{J}_{adv} = \left(\rho c_p\right)_f \boldsymbol{u}. \tag{18}$$

There may also be heat sources f, either as a consequence of direct heating or cooling of the fluid and porous medium of the cell or of a flow well introducing or extracting fluid of a given temperature. Accounting for the heat exchange from the rock to the fluid through the term $\boldsymbol{j}_{rf}$, we obtain the energy equation for the fluid

$$\phi\left(\rho c_p\right)_f \frac{\partial T_f}{\partial t} + \left(\rho c_p\right)_{\boldsymbol{f}} \boldsymbol{u} \cdot \nabla T_f - \nabla \cdot \left(\boldsymbol{\kappa_f} \nabla T_f\right) - \boldsymbol{j}_{rf} = f_f, \tag{19}$$

where subscripts *f* indicate the fluid and the porosity factor of the first term takes into account that the fluid only takes up part of the total volume. The heat may also change in and propagate through the rock itself. As the rock is immobile, the corresponding equation is as for the fluid, but without the advection term and reversed sign of the rock fluid exchange term

$$(1-\phi)(\rho c)_r \frac{\partial T_r}{\partial t} - \nabla \cdot (\boldsymbol{\kappa}_r \nabla T_r) + \boldsymbol{j}_{rf} = f_r. \tag{20}$$

As long as the flow is reasonably slow, one can assume local thermodynamic equilibrium, meaning that the temperatures of rock and fluid are equal: $T_r = T_f = T$ within each REV. Then we can add the two equations obtaining the overall heat equation

$$(\rho c)_{eff} \frac{\partial T}{\partial t} + \left(\rho c_p\right)_f \boldsymbol{u} \cdot \nabla T - \nabla \cdot \left(\boldsymbol{\kappa}_{eff} \nabla T\right) = f, \tag{21}$$

using the effective heat capacity per volume, thermal conductivity and total heat sources defined by

$$(\rho c)_{eff} = (1-\phi)(\rho c)_r + \phi\left(\rho c_p\right)_f, \tag{22}$$

$$\boldsymbol{\kappa}_{eff} = (1-\phi)\boldsymbol{\kappa}_r + \phi \boldsymbol{\kappa}_f, \tag{23}$$

$$f_{eff} = f_f + f_s. \tag{24}$$

In a purely numerical study such as this, it is useful to reduce the number of parameters and constants to the minimum. By so doing, we arrive at a set of reduced parameters which determine the behaviour of the system in the mathematical model. If we assume the heat conductivity to be isotropic, this can be done by dividing Equation 21 by $\left(\rho c_p\right)_f$ and defining

$$\alpha = (\rho c)_{eff} / \left(\rho c_p\right)_f \quad \boldsymbol{\kappa} = \boldsymbol{\kappa}_{eff} / \left(\rho c_p\right)_{eff} \quad f = f_{eff} / \left(\rho c_p\right)_{eff}, \tag{25}$$

yielding

$$\alpha \frac{\partial T}{\partial t} + \boldsymbol{u} \cdot \nabla T - \boldsymbol{\kappa} \Delta T = f. \tag{26}$$

We will assume the new storage term constant α to equal one throughout. This leaves us with a single parameter describing the relative importance of advective and diffusive transport. Because we are primarily interested in studying flux fields, we set $\boldsymbol{\kappa} = 0$ in all but one test case. In that case, the equation becomes a linear transport equation, with temperature as a passive tracer advected in the flow field.

2.5 Fractures

In certain cases, some features may not satisfy the conditions allowing us to use the averaging REV approach, and must be accounted for explicitly. This is true for what we call fractures of the rock. They are larger than the "other" pores and their properties may differ dramatically from

those of the rest of the rock. Consequently, they have a great impact on the fluid flow and we should account for them in our models. Although strict scale separations do not in general exist for realistic scenarios, we will assume that we are able to determine which features to resolve explicitly.

Fractures can be open and enhance flow or closed, blocking it. Be their effect on the flow within some limit, one can upscale it and include it in a modified permeability of the porous medium, as done in the continuum models. If the effect on the flow is large enough and too statistically heterogeneous for upscaling, we have little choice but to model them explicitly given that they are too large for averaging. If the fractures are highly permeable, one might be tempted to assume that all the flow takes place in the fractures and neglect the surrounding porous medium altogether. This is the approach taken in the so-called Discrete Fracture Network models. While reasonable in cases of extreme permeability ratio, the assumption may not apply in the more moderate permeability ratio range (Matthäi & Belayneh, 2004).

In the intermediary permeability ratio scenarios, a combination of the two approaches will usually be the most accurate. One includes the effect of all small-scale fractures in the permeability of the porous rock matrix, but models the larger fractures explicitly. Note that in some cases, the rock itself is practically impermeable if the small-scale fractures are neglected. Then, the permeability of the matrix is determined from the small fractures alone. Both the methods presented in this thesis belong to this class of Discrete Fracture Matrix models.

A further defining characteristic is that they are very thin compared to the length. So thin, in fact, that they are often modelled in codimension one, i.e., two-dimensional planes and one-dimensional lines in a three- and two-dimensional domain, respectively. This is the approach taken in both the flow methods discussed in this work.

All cases investigated in this work are in two dimensions. This means the fractures are exceedingly thin lines, and their intersections approach zero-dimensional points. When we discretize the domain in Chapter 3, we will indeed consider the fractures and intersections as one- and zero-dimensional, respectively.

We mentioned in Section 2.2 that the assumptions for Darcy's law might be invalid in open fractures. In this case, the common approach is to model the flow along the fractures using what is referred to as the "cubic law". We start out from the incompressible Navier-Stokes equations in a domain bounded by two parallel lines of length L and separated by an aperture a (see Figure 3). Ignoring gravity, we have

$$\rho\left(\frac{\partial \boldsymbol{u}}{\partial t} + \boldsymbol{u}\cdot\nabla\boldsymbol{u}\right) = -\nabla p + \mu\Delta\boldsymbol{u}. \tag{27}$$

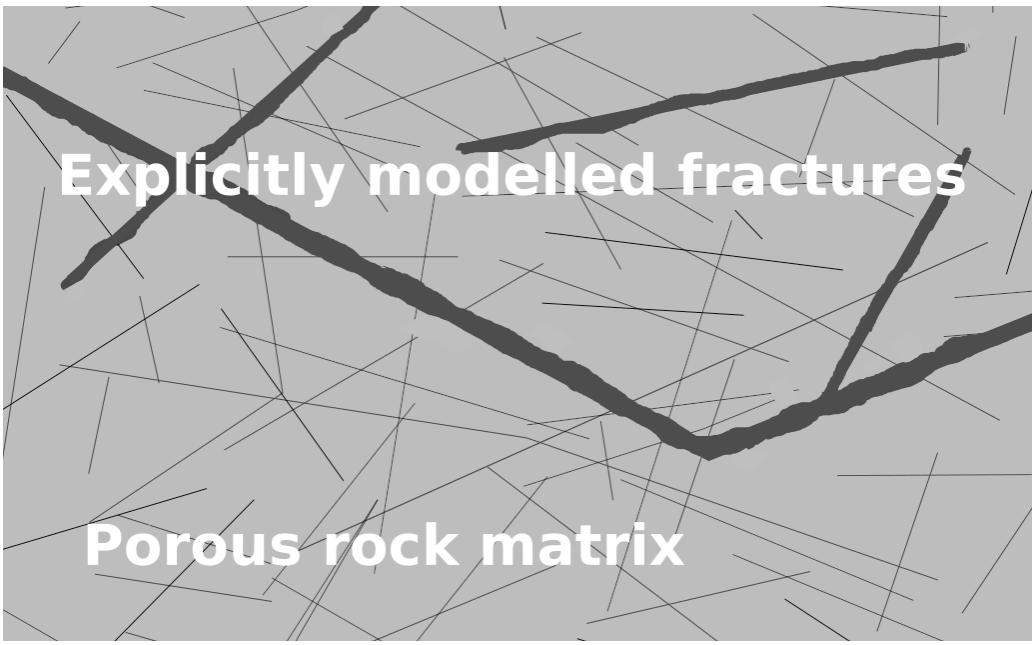


Figure 2: A porous medium with several small fractures and a couple of larger ones. The former are accounted for by the matrix permeability in our models, the latter are modelled explicitly.

Further, we assume steady state and set pressure conditions $p_1 > p_2$ at the two boundaries separated by L and no-slip boundaries at the top and bottom. Ignoring body forces, in particular gravity, we are left with

$$\frac{\partial^2 u_x}{\partial y} = \frac{p_2 - p_1}{\mu L}. \tag{28}$$

Integrating twice and imposing the boundary conditions, we arrive at the velocity profile across the fracture, which we use to evaluate the average velocity:

$$u_x(y) = \frac{1}{2\mu}\frac{p_1 - p_2}{L}\left[\left(\frac{a}{2}\right)^2 - y^2\right] \tag{29}$$

$$\Downarrow$$

$$\bar{u}_x = \frac{1}{a}\frac{1}{2\mu}\frac{p_1 - p_2}{L}\int_{-\frac{a}{2}}^{\frac{a}{2}}\left[\left(\frac{a}{2}\right)^2 - y^2\right]dy = \frac{a^2}{12\mu}\frac{p_1 - p_2}{L}. \tag{30}$$

If we introduce a modified permeability $\boldsymbol{K}^* = \frac{a^2}{12}$, the average velocity has the form of an Equation 12 Darcy velocity, since $\frac{p_1 - p_2}{L}$ is the average pressure gradient. Note that the average flux in Equation 30 implies a total volumetric flux of

$$q_V = t\frac{p_1 - p_2}{L} = \frac{a^3}{12\mu}\frac{p_1 - p_2}{L} = t\frac{p_1 - p_2}{L}. \tag{31}$$

The transmissivity t equals the product of the permeability and the cross-sectional area, or just the aperture in two dimensions. So the total volumetric flux is proportional to the *cube* of the aperture, hence the name "cubic law" *(Zimmermann & Böðvarsson, 1994)*.

Naturally occurring fractures are, obviously, never bounded by ideal parallel plates. The effects of wall roughness and variations in width are usually accounted for by introducing a hydraulic aperture a_{hyd} in Equation 30. Throughout this thesis, we will assume it to equal the geometrical aperture and to be constant for each fracture. Moreover, the fracture permeabilities $\boldsymbol{K}_f$ used in Section 4 are assumed to be the modified $\boldsymbol{K}^*$.

The fracture intersections play a critical role in providing the interconnectivity between the fractures. In our models, they generally inherit the parameters, here permeability, of the intersecting fractures. A complication arises if the fractures in question have different permeabilities. Considering that the fracture formation has a geological history, one approach would be to assign the permeability of the youngest fracture to the intersection. Where historical information or other data which could guide the choice is hard to come by, one is left with the option of assigning some average value. Consider, however, that the choice may seriously influence the flow, especially in the more permeable fractures. There, the intersection permeability determines whether or not the flow is blocked, as we will see in several of the Section 4 simulations. On the other hand, a short segment of higher permeability will not alter the flow along a low-permeable fracture significantly. The same considerations apply in principle to the heat conductivity of the intersections. But because of the $\phi\kappa_f$ term in Equation 23 and the conductivity value range of common porous media and fluids, the thermal conductivity will very rarely change as drastically as the permeability. Therefore, the choice of intersection conductivities is not as critical as that of permeabilities.

In fractures with permeability on the same order as the matrix's or lower, Darcy's law can be applied directly if it applies in the matrix. We assume that all other parts of our model apply to all of the fractures as well as the surrounding rock.

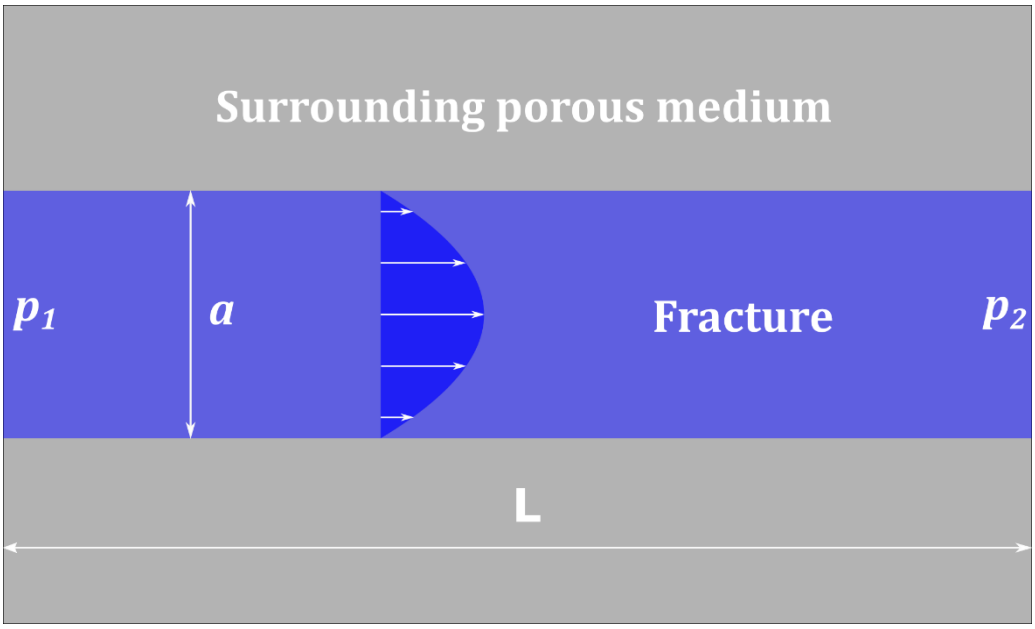


Figure 3: A simplified open fracture in a porous medium of aperture a and length L. The velocity profile resulting from the assumption of constant pressures at the ends of the fracture and no-slip conditions along it is shown in the middle.

2.6 Closing the System

Suppose we want to solve the two Equations 12 and 26 in some two-dimensional domain Ω restricted by the boundary $\partial\Omega$. Then, boundary conditions are necessary to close the system of partial differential equations. We restrict ourselves to two types, namely Neumann conditions given by a function g_N specifying the flux over the borders of the domain and Dirichlet conditions fixing the primary variables pressure or temperature to equal some g_D on the boundary. Dividing the boundary into the non-overlapping $\partial\Omega_N$ and $\partial\Omega_D$ and denoting the outward normal vector of the domain by $\boldsymbol{\nu}$, they can be written

$$\begin{aligned} \boldsymbol{u}\cdot\boldsymbol{\nu} &= g_{N,p} \qquad && on\ \partial\Omega_{N,p} \\ p &= g_{D,p} \qquad && on\ \partial\Omega_{D,p} \end{aligned} \tag{32}$$

for the pressure equation and similarly for the heat flux $\boldsymbol{J}$ and heat T. There is nothing preventing us from imposing boundary conditions of different type for the two equations, i.e., $\partial\Omega_{N,p}$ is not necessarily equal to $\partial\Omega_{N,T}$.

If there are no Dirichlet conditions, the solution of the system is still only determined up to a constant, i.e., for a given solution $\hat{p}$ all solutions on the form $p = \hat{p} + p_c$ with p_c a constant are also valid solutions. In this case, the solution is fixed by prescribing an interior pressure through a source or sink, as done in Section 4.4.

In the case of the transient heat equation, we also need to set an initial condition. This is a

description of the heat distribution at the beginning of the simulation on the form

$$T = T_0 \quad at\ t = 0. \tag{33}$$

We can now sum up the entire system in the following set of equations

$$\begin{aligned}
\nabla \cdot (\boldsymbol{K}\nabla p) - s &= 0 && in\ \Omega, \\
\boldsymbol{u} \cdot \boldsymbol{\nu} &= g_{N,p} && on\ \partial\Omega_{N,p}, \\
p &= g_{D,p} && on\ \partial\Omega_{D,p}, \\
\frac{\partial T}{\partial t} + \boldsymbol{u} \cdot \nabla T - \boldsymbol{\kappa}\Delta T &= f && in\ \Omega, \\
\boldsymbol{J} \cdot \boldsymbol{\nu} &= g_{N,T} && on\ \partial\Omega_{N,T}, \\
T &= g_{D,T} && on\ \partial\Omega_{N,T}, \\
T &= T_0 && in\ \Omega\ at\ t = 0.
\end{aligned} \tag{34}$$

For the remainder of this thesis, we will assume all quantities to be scaled, so that the system is dimensionless. This is done because the work is purely numerical, and so we are more interested in investigating behaviour for different parameter ratios, e.g. between fracture and matrix permeability, rather than their individual values.

3. Numerical Methods

The system of equations derived in the previous chapter is solved using numerical methods. Applying the assumptions of introduced at the beginning of Section 2.4 allows us to do the simulations sequentially, i.e., we first solve the flow problem and then the heat transport. By introducing discrete variables and manipulating the system, one arrives at a system of linear equations suited for computer solving. In the following chapter, we will first provide some general definitions and comments on discretizations. Then we proceed to describe the two flow methods. Emphasis will be placed on how the fractures are modelled. A section on temporal discretization and the discretization of the transport equation is succeeded by a description on how intersection cells may be eliminated and a summary of how the methods may be combined and their implementation. We conclude the chapter by describing our tools for evaluating the methods.

3.1 Gridding

The problems are solved in some spatial domain Ω. As mentioned in the previous chapter, we denote its border by $\partial\Omega$. Because we model the fractures as lower-dimensional, it is fruitful to divide the domain hierarchically in subdomains. As all the examples presented in the following chapters are two-dimensional, we will for the sake of clarity concentrate on that case.

Adapting the notation of (Boon & Nordbotten, n.d.), we have a two-dimensional domain Ω^2 with subdomains ${\Omega_i}^2$ representing the matrix. The fractures are represented by the union of one-dimensional domains $\Omega^1 = \bigcup {\Omega_i}^1$ and intersections by $\Omega^0 = \bigcup {\Omega_i}^0$. For an illustration, see the top left Figure 4, where we have three subdomains of Ω^2 separated by the fractures. The three fractures themselves make up three subdomains of Ω^1 and the one intersection present constitutes Ω^0.

When we want to solve the problem defined above numerically, we first of all discretize in space. This means we partition the domain in a finite set of non-overlapping subdomains, called cells. The general idea is to associate to each cell a certain number of variables, or computational degrees of freedom, to be determined when the system is solved. Before we describe how this is done for each method, we give some definitions and general properties related to the geometry of the discretization or grid.

In all grids used herein, we use triangles for the <u>cells</u> of the matrix, i.e., the non-fracture part of the domain. In the context of the second flow method, the cells are often referred to as elements. We could have used quadrilaterals as well, and indeed a Cartesian all-orthogonal grid would simplify the geometrical aspects of the methods. But we want to grid the domain in such a way that the fractures always lie on the <u>faces</u> between two cells. This approach results in what is

called *conforming grids*, where we represent the fractures in the model by placing lower-dimensional cells along them. A Cartesian grid would prevent us from resolving complex and arbitrary fracture geometries explicitly.

In two dimensions, a face is the line between two neighbouring cells. In other words, the fractures are one-dimensional in our grids. The endpoints of the faces are called the nodes. The full partition into triangles, or triangulation, is uniquely defined by the coordinate list of these points and the so called connectivity list, a $(\text{number of triangles}) \times 3$ matrix where each row contains the indexes of the three corner points of one cell.

For the method to be sound, we require that it converges to the true solution as we decrease the cell size. There is much general analysis on this subject, and it tells us that we should seek a discretization which respects the following:

- The cells should not approach degenerate shapes, i.e., avoid triangles where one of the faces is much shorter than the other two.
- The faces should be unique, in the sense that any face should make up the entire side of both neighbouring cells. This implies that the cells along each fracture have length equal to the face of the two adjoining matrix cells, as indicated by the green fracture-fracture faces of Figure 4.
- The size of the cells should not vary too much. This will be reflected by the condition number of the solution matrix, which tells us how accurately we can hope to solve the system. However, small cells are inherent to the fractures because of their thinness. In particular, the cells at the intersections of two fractures will have a size proportional to the square of the fracture aperture and consequently be orders of magnitude smaller than the normal cells. Therefore, it can be tempting to somehow rid oneself of these cells by some elimination procedure. Some possible approaches are described in the next subsections.

A grid generation algorithm which honours these goals is the so-called Delauney triangulation. Simply speaking, it demands that the circle defined by the three points of a triangle contains no other points than those three. Thereby, the maximum and minimum angle of the grid are minimized and maximized, respectively, and the cells have reasonably evenly sized faces. For more details on unstructured grids and desired properties, we refer to (Liseikin, 1999).

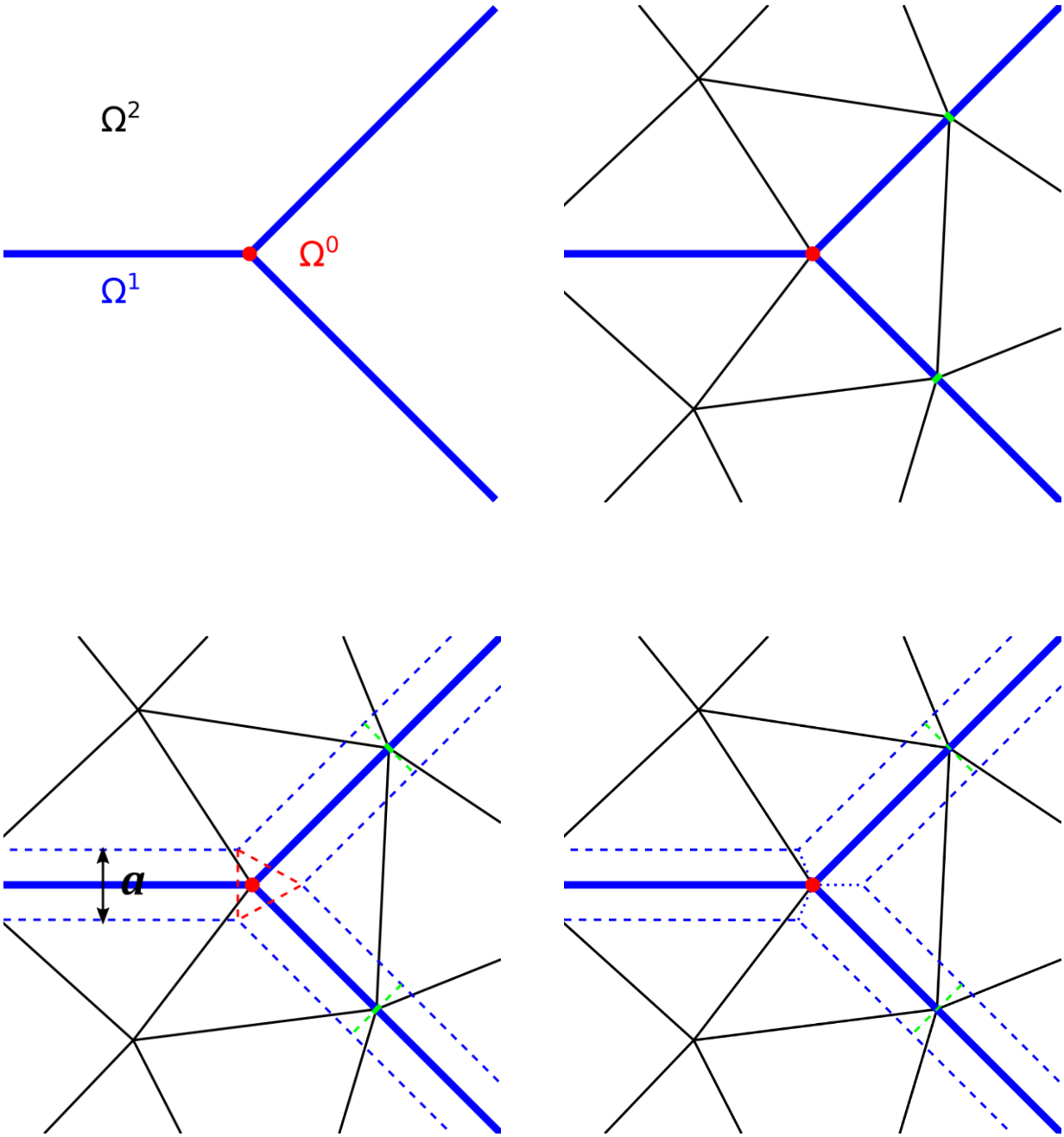


Figure 4: For the domain geometry top left, we show a grid example at the top right and the adjustments made for the computational domain for the Finite Volume method at the bottom. Fractures are shown in blue, intersections in red, matrix faces in black and fracture-fracture faces in green.

In addition to being demanding merely because of the number of fractures to be accounted for, realistic fracture network geometries include components which for different reasons may challenge the model and its implementation. We identify the following characteristics, which may be combined:

- Single, isolated fractures.
- Connected fracture networks.
- Fractures ending inside the domain or at the boundaries.
- Intersections of multiple fractures, in two dimensions most commonly in the form of X (four-way) or Y (three-way) intersections.
- Fractures which cannot be described as a single straight line. These knickpoints might also be seen as L intersections.
- Intersections including small angles.
- Small distances between unconnected fractures.

The features in the second list may challenge the principles in the first. But while especially close distances and small angles may lead to awkwardly shaped or sized cells unless special care is taken, we did not encounter any test case where we were not able to produce conforming grids and run simulations with satisfactory outcome. While the two first geometries of Section 4 only include some of the geometrical components listed, the last one includes all but the knickpoints, which are included in the second geometry, see e.g. the fracture distribution maps at the top left of Figure 21 and Figure 27.

3.2 Cell Centred Finite Volume Method

After the domain has been gridded as described in the previous section and shown at the top of Figure 4, discrete variables are assigned to each cell. We describe the primary variable pressure as constant in each cell, and formulate the model also in terms of the secondary variable flux by defining one flux for each face of the grid. We will now describe how the control volume finite difference method arrives at a set of linear equations to solve.

We start out from the form with surface integrals, Equation (6), and as above assume incompressibility of the fluid. We enforce mass conservation for each discrete cell i by replacing the surface integral by a summation over the m_i faces of the cell and obtain one equation for each cell of the form

$$\sum_{f=1}^{m_i} A_f \boldsymbol{n}_f \cdot \boldsymbol{u}_f = s_i, \tag{35}$$

where A_f and $\boldsymbol{n}_f$ are the area and unit normal vector of the face *f*, $\boldsymbol{u}_f$ is the flux over that face and s_i is the source contribution in cell *i*. We denote the left hand side product by, the discharge over the face between cells *i* and *j*, by u_{ij}.

To compute the discharges, we use a two-point flux approximation (TPFA) based on the cell-centre pressures

$$u_{ij} = T_{ij}(p_i - p_j), \tag{36}$$

where p_i and p_j are the pressures in the neighbouring cells and T_{ij} is the face transmissibility. The latter accounts for permeability and geometry, and is computed from the two half transmissibilities corresponding to the face f between the two cells as

$$T_{ij} = \frac{\alpha_{i,f}\alpha_{j,f}}{\alpha_{i,f}+\alpha_{j,f}}. \tag{37}$$

The half transmissibility of face *f* of cell *i* is in turn given as

$$\alpha_{i,f} = \frac{A_f \boldsymbol{n}_f \cdot \boldsymbol{K}_i}{\boldsymbol{d}_f \cdot \boldsymbol{d}_f} \boldsymbol{d}_f, \tag{38}$$

where $\boldsymbol{K}_i$ is the permeability assigned to the cell, $\boldsymbol{d}_f$ is the distance vector from cell centre to face centroid and $\boldsymbol{n_f}$ is the unit normal vector pointing outward from cell i.

Complementing the unknown pressures at the centroids of the matrix cells, there are unknowns associated to the centroids of the fracture cells. As mentioned above, the domain is partitioned with fractures, the blue lines in Figure 4, coinciding with the interior faces between matrix cells. These faces are converted into fracture cells as they are assigned apertures a, which multiplied with the original face length give the volume of the new cells. The aperture is also used to produce hybrid faces for the matrix-fracture interfaces (dashed blue lines). These faces, parallel to the fracture but displaced half an aperture to either side, enable us to compute the half transmissibilities corresponding to these faces. The procedure is as for the matrix-matrix connections, with a small adjustment to the distance vectors. On the fracture side of the face, we use a vector of length a perpendicular to the fracture. On the matrix side, this vector is subtracted from the $\boldsymbol{d}_f$ defined above. Where a fracture ends within the domain, we assume the flow to be negligible, as the fracture tip is orders of magnitude smaller than the tangential boundary of the fracture.

By default, the intermediate fracture intersection cells shown in red in Figure 4 are excluded, leading to direct coupling of the fracture cells neighbour to the intersection cell. Although the direct connections are visualized by blue dotted lines, these connections should not be considered regular faces, but rather as computational links between the cells and their unknowns. The purpose of the removal is both to obtain a smaller condition number and to avoid severe time-step restrictions associated with small cells. To each new face between cell i an j, we assign face transmissibilities calculated using the star delta transformation as described by *(Karimi-Fard, et al., 2004)*:

$$T_{ij} = \frac{\alpha_i \alpha_j}{\sum_k^n \alpha_k}. \tag{39}$$

The αs are again half transmissibilities and n is the number of fracture cells meeting at the intersection. However, the implementation we have used also includes the option to keep the intersection cells. Indeed, the advisability of the intersection cell removal is one of the main questions of this thesis. and we will return to this point in much more detail in Sections 3.6 and 4.

Boundary conditions of Dirichlet or Neumann type are set at the domain boundary and wherever fractures run all the way to the boundary, and collected in a right hand side vector $\boldsymbol{b}$**.** Then we collect the left hand side factors multiplied with the pressure in type 35 equations for all cells in a solution matrix **A**. Denoting the pressure solution vector by $\boldsymbol{x}$**,** we can now solve an equation system on the linear form

$$\mathbf{A}\boldsymbol{x} = \boldsymbol{b}. \tag{40}$$

An alternative to the TPFA is the *multi* point flux approximation, where several of the surrounding cells are connected for the computation of each face flux instead of just the two immediate neighbours. While requiring a somewhat more involved implementation and being computationally more expensive, it is significantly more robust. The MPFA's main advantage is that unlike the TPFA it is applicable to cases of anisotropic permeability fields. As our cases do not include this feature, we choose to use the simpler TPFA version, and comment at the one error we observe which would be improved by switching to MPFA. However, the implementation we use includes both approximations, and we refer to Sandve et al (2012) for further details. Particulars on the generalization to three dimensions may also be found therein.

3.3 Mixed Finite Elements

The second approach is a recent method which is based on the mixed finite element and mortar methods and presented in (Boon & Nordbotten, n.d.). The mass conservation and Darcy's law are applied directly in the matrix in the form of Equation 12, leading to the system of the three first lines of Equation 34. When we use finite elements, we choose a set of basis functions for our unknowns, and then seek weights for the basis functions satisfying the system of equations in our domain. In the mixed finite element, local mass conservation is obtained by including flux unknowns at the faces as well as unknown pressures.

The mortar method is a technique for decomposing the domain. It may be exploited to perform different discretizations in the different subdomains according to local variations in solution behaviour and to mesh the domain in a non-matching way (Bernardi, et al., 2005). In our description of how it is combined with the mixed finite elements in the particular method used in this thesis, we continue to use the Section 3.1 geometry notation and point to Figure 5 for a visualization of some of the central concepts. Denoting the dimension by superscripts d, each lower dimensional subdomain ${\Omega_i}^d$ has a boundary ${\Gamma_i}^d$ to the surrounding domain Ω^{d+1}.

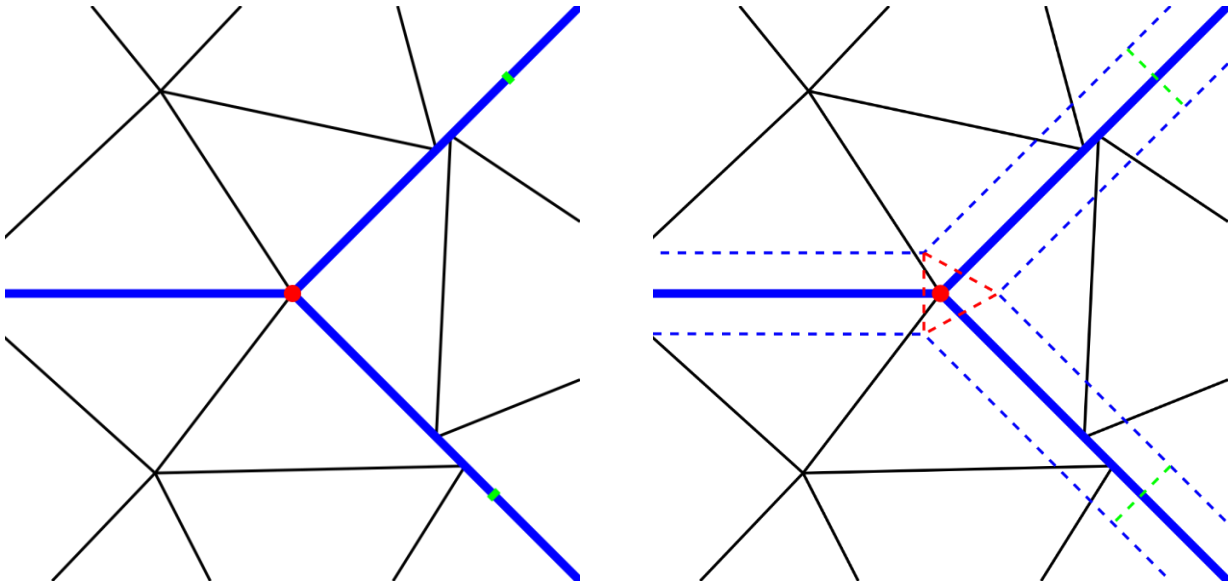


Figure 5: Geometrical and computational grid for the Mixed Finite Element method. The grid is non-matching, i.e., nodes of matrix cells do not necessarily match across the fractures.

We now concentrate on the lower dimensional subdomains, the intersections and fractures corresponding to $d = 0$ or 1. Here, a few additional components are required for the coupling between dimensions. We introduce mortar variables λ_{ij} which represent the discharge from the normal fluxes on both sides of the fracture, i.e., $\lambda_{ij} = \boldsymbol{u} \cdot \boldsymbol{\nu}$ for the flux over the *j*th border Γ_{ij}^{d} of subdomain Ω_i^d. For a subdomain with γ_i such borders, the difference in normal fluxes is evaluated using a jump operator

$$[\![\cdot]\!] : L^2(\Gamma^d) \to L^2(\Omega^d), \qquad [\![\lambda]\!] = -\sum_{j=1}^{\gamma_i} a_j^{d+1} \lambda_{ij} \,. \tag{41}$$

Then, again denoting the aperture by *a*, the mass conservation for a fracture (intersection) domain becomes

$$[\![\lambda]\!] + a\nabla \cdot \boldsymbol{u} = f, \tag{42}$$

where f and $\boldsymbol{u}$ are the source term and tangential, *d*-dimensional flux within the fracture, respectively. In the one-dimensional fractures, we let K^1 denote the tangential permeability tensor in the fracture and note that permeability scales with the aperture, i.e., $K^1 = aK$ for a tangential permeability independent of the aperture. Thus, Darcy's law within the fracture is

$$\boldsymbol{u} = -aK\nabla p. \tag{43}$$

Where a fracture ends at the domain boundary, we apply the same boundary conditions as are given to the corresponding part of the boundary for the matrix. Fractures ending within the domain are given a no-flow Neumann condition, as in the other method. In addition to these, we need conditions for the coupling of the tangential fluxes between a *d*-dimensional subdomain (fractures and intersections) and its surrounding *d+1*-dimensional domain; fractures immersed in the matrix or intersections and their bordering fractures. Analogously to the effective permeability approach described in Section 2.5, we introduce a scaling parameter $\alpha = [0,1]$ for the dependency of the permeability on the aperture and the normal permeability K_ν . With the traces of the surrounding pressures p_j, Darcy's law for the flow across the fracture boundary Γ_{ij} of Ω_i becomes

$$(\lambda_{ij})\frac{a}{2} = -a^{\alpha} K_\nu (\mathrm{p} - p_j). \tag{44}$$

For the weak form, this coupling requires an extension operator $\mathcal{R}$ from the mortar space to

the surrounding domain. Then, the total flux can be represented by the sum of a function with zero normal flux and this extension $\boldsymbol{u} = \boldsymbol{u}_0 + \boldsymbol{\mathcal{R}\lambda}$.

We define the spaces

$$\mathbf{V}^d = H(div; \Omega^d), \quad \Lambda^d = \left(L^2(\Gamma^d)\right), \quad Q^d = L^2(\Omega^d) \tag{45}$$

for the (tangential) fluxes, (normal) mortar fluxes and pressures, respectively, and the corresponding $\mathbf{V}_0^d$ for the zero normal fluxes. Then, composite function spaces are introduced to enable a unified treatment of the domains:

$$\boldsymbol{\mathcal{V}}_0 = \oplus_{d=1}^{\mathbf{2}} \boldsymbol{V}_0^d, \quad \Lambda = \oplus_{d=0}^{\mathbf{1}} \Lambda^d, \quad \mathcal{Q} = \oplus_{d=0}^{\mathbf{2}} Q^d. \tag{46}$$

Multiplying the above derived equations with the appropriate test functions and integrating, (Boon & Nordbotten, n.d.) arrive after some manipulation at the weak formulation of the problem:

We seek $(\boldsymbol{u}_0, \lambda, p) \in \boldsymbol{\mathcal{V}}_0 \times \Lambda \times \mathcal{Q}$ satisfying

$$\begin{aligned}
&(K^{-1}(\boldsymbol{u}_0 + \mathcal{R}\lambda), \boldsymbol{v}_0)_\Omega - (ap, \nabla \cdot \boldsymbol{v}_0)_\Omega = -\langle g, \boldsymbol{v}_0 \cdot \boldsymbol{\nu}\rangle_{\partial\Omega_D}, \ \ \forall \boldsymbol{v}_0 \in \boldsymbol{\mathcal{V}}_0, \\
&(K^{-1}(\boldsymbol{u}_0 + \mathcal{R}\lambda), \mathcal{R}\mu)_\Omega - (ap, \nabla \cdot \mathcal{R}\mu)_\Omega + \\
&\qquad a^{1-\alpha}\frac{{K_n}^{-1}}{2}\langle \lambda, \mu\rangle_\Gamma - (\mathrm{p}, [\![\lambda]\!])_\Omega = 0, \qquad \forall \mu \in \Lambda, \\
&-(\nabla \cdot (\boldsymbol{u}_0 + \mathcal{R}\lambda), aq)_\Omega - ([\![\lambda]\!], q)_\Gamma = -(f, q)_\Omega, \qquad \forall q \in \mathcal{Q}.
\end{aligned} \tag{47}$$

Finally, one has to choose the discrete spaces one wishes to use. In this work, we have used the lowest order possible, as also described in (Boon & Nordbotten, n.d.). This corresponds to piecewise constants for the pressures and mortar spaces and lowest order Raviert-Thomas space for the flux space. Again denoting the solution vector, here including both pressures and fluxes, by $\boldsymbol{x}$ and collecting the known boundary conditions and source terms in the vector $\boldsymbol{b}$**,** the equation system 47 left hand side can be organised in matrix form again yielding an equation on the linear equation 40 form. Of course, the form of both $\mathbf{A}$ and $\boldsymbol{b}$ will in general not be the same for the MFE system as for the FV system.

As mentioned at the start of this chapter, the mortar treatment of domains of different dimensionality allows for grids which do not match at opposing sides of a fracture, see Figure 2. Notice that the green fracture-fracture faces to the left and the same faces depicted in dotted

green lines to the right are also moved; the fracture cells are longer than the faces of the adjoining matrix cells.[1] For certain geometries, this can be a significant advantage in terms of number and quality of elements. Where desirable, we are even able to circumvent the usual coupling requirement that the fracture grid be no finer than the surrounding matrix grid by choosing a richer function space for the fractures. Another possibility is to include more complex physics in the fractures. For example, if the fractures are highly permeable, Darcy's law might not hold. In such cases, one could keep the same model for the matrix, but use other models in the fractures.

3.4 Upwind Discretization

We discretize the heat transport equation using a finite volume approach. Here, we cannot assume the storage term $\frac{\partial T}{\partial t}$ to vanish as we did in the mass conservation. As can be seen by comparing Equations 26 and 12, the diffusion term and the Darcy fluxes appear in exactly the same mathematical form, with heat diffusivity replacing permeability. Therefore, we account for diffusion as described for the fluxes in Chapter 3.2. Specifically, we calculate a heat diffusivity D_{ij} as we described for the transmissibilities T_{ij}, only replacing the permeability value $\boldsymbol{K_i}$ of Equation 38 by the cell conductivity $\boldsymbol{\kappa}_i$.

We have chosen to discretize the advective term using the simplest form of upwinding. The idea is to exploit that we know the direction of the fluxes. Thus, we would expect the amount of heat advectively transported over the face *f* between cells *i* and *j* to be determined by the heat in the cell where the flux originates, and we model the advective flux as

$$\boldsymbol{J}_{adv,f} = \boldsymbol{u}_f T_{upw,ij}, \tag{48}$$

where

$$T_{upw,ij} = \begin{cases} T_i & if \quad \boldsymbol{u}_f \cdot \boldsymbol{n}_f > 0 \\ T_j & i\,f \quad \boldsymbol{u}_f \cdot \boldsymbol{n}_f < 0 \end{cases} \tag{49}$$

and $\boldsymbol{u}_k \cdot \boldsymbol{n}_f > 0$ means the flow is from cell *i* to *j* and vice versa. This leads to the semi-discrete form of Equation 26

$$\frac{\partial T_i}{\partial t} + \sum_{f=1}^{m_i} \boldsymbol{u}_f T_{upw,ij} - D_{ij}\left(T_j - T_i\right) = f_i. \tag{50}$$

[1] Note that this violates the general gridding guideline that the faces should be unique, cf. Section 3.1, and is only permitted because of the special mortar coupling between domains of different dimensionality.

As stated in Section 2.6, we have included two kinds of boundary conditions, Neumann and Dirichlet, whose values we denote by g_N and g_D, respectively. The discrete mathematical form of the flux over a Neumann boundary face f on cell i is

$$\boldsymbol{J}_{N,f} = g_{N,f}$$

and can be assigned directly in the equation of the cell. Given the flux $\boldsymbol{u}_f$ on a Dirichlet face, the heat flux is

$$\boldsymbol{J}_{D,f} = \boldsymbol{u}_f T_{upw,D} - D_f(g_D - T_i), \tag{51}$$

where D_f is the diffusivity of the face and the boundary upwind heat value is

$$T_{upw,D} = \begin{cases} T_i & if \quad \boldsymbol{u}_f \cdot \boldsymbol{n}_f > 0 \\ g_D & if \quad \boldsymbol{u}_f \cdot \boldsymbol{n}_f < 0 \end{cases}. \tag{52}$$

In addition to this, we allow for interior well contributions regarded as heat fluxes. As such, they can be added directly to the equation for the cell in which they occur in much the same way as Neumann boundary conditions.

As mentioned above, this thesis is primarily a comparison study of different methods for flow. As such, the heat transport serves just as much as a means of revealing properties of the methods presented above and the fluxes they produce. Therefore, we will in most cases assume that there is no diffusion ($\kappa = 0$), as this term will obscure the information about the flow field. This is not to say that the insights gained may not be applicable to cases where diffusion plays a role, as argued in Section 4.3.

The artificial volume added to the domain by assigning fracture apertures is hardly relevant when calculating the pressures as we correct for the aperture in the transmissibility computation. However, it might in some cases influence the temperature solution slightly, because the latter is an extensive variable, which will be "diluted" over a larger domain than it should. The assumption is that this can be neglected because of the thinness of the fractures. In the most realistic of the cases presented in Section 4, for example, the fracture area constitutes less than $3 \cdot 10^{-4}$ of the entire domain. The argument is repeated when we remove the intersection cells. They are on the order of a^2, and consequently make up a comparable fraction of the total fracture area.

3.5 Temporal Discretization of Transport

We also have a time derivative in Equation 50, and this presents us with the choice of temporal discretization. For this thesis, we mainly use the standard explicit discretization where the new

value of the variables is calculated directly from the values of the previous time steps. Using superscripts *n* and *n+1* for the two successive time steps and Δt for the length of the time step, we write

$$\frac{T_i^{n+1} - T_i^n}{\Delta t} + \sum_{j=1}^{m_i} \boldsymbol{u}_k T_{upw,ij}^n - D_{ij}\left(T_j^n - T_i^n\right) = f_i^n \text{ or} \tag{53}$$

$$T_i^{n+1} = T_i^n - \Delta t \left(\sum_{j=1}^{m_i} \boldsymbol{u}_k T_{upw,ij}^n - D_{ij}\left(T_j^n - T_i^n\right) - f_i^n \right). \tag{54}$$

This equation will be applied to every cell and for every interior face in our spatial domain once for each time step we wish to perform.

A discretization scheme as in Equation 6254 is not unconditionally stable. By this, we mean that unless certain requirements are met, the numerical solution will diverge from the true solution in the sense that a small perturbation of the data will be amplified for each new time step taken. To analyse the stability, we first examine the purely advective case ($\kappa = 0$) in one dimension. Suppose we solve a problem of uniform flow of positive velocity u on an equidistant grid, as sketched in Figure 6. The exact solution at any point x_1 at time $t_1 = t_0 + \Delta t$ is identical to the solution at the upstream point $x_0 = x_1 - u\Delta t$. For the numerical scheme to capture this, it must be able to transport information at least as quickly as the flow it simulates. Thus, an upper bound on the allowed time step in terms of the velocity and cell spacing Δx arises. We formulate this in the CFL condition named after Courant, Friedrichs and Lewy

$$\Delta t \leq \frac{\Delta x}{u} \quad \text{or} \quad u\frac{\Delta t}{\Delta x} \stackrel{\text{def}}{=} CFL \leq 1. \tag{55}$$

If this criterion is violated, the maximum discrete velocity of the numerical method is smaller than the flow velocity and the numerical scheme is unstable. In the case sketched in Figure 6, where $|\boldsymbol{u}| = 2\Delta x$, the CFL condition reads $\Delta t = 1/2$.

Similarly, von Neumann stability analysis tells us that there is also a time step restriction originating from the diffusive term. In the one-dimensional case, for a model equation without any advection, it reads

$$\Delta t \leq \frac{\Delta x^2}{\kappa}. \tag{56}$$

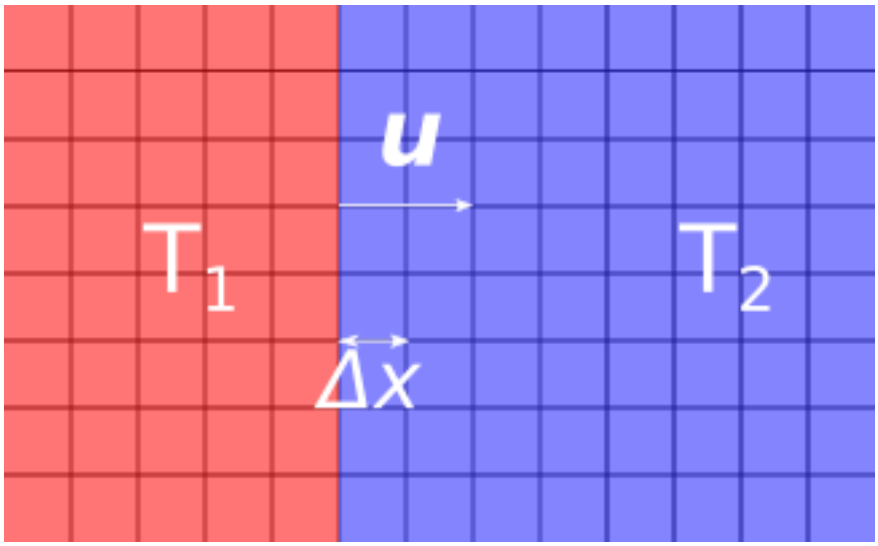


Figure 6: A uniform flow left to right transports the temperature T_1 into the domain. We also show an equidistant Cartesian grid of cell size Δx.

Our simulations will neither be in one dimension nor on regular, equidistant grids, leading to a somewhat more complicated time step criterion. In addition to depending on the numerical scheme and flow velocity, it will also vary according to cell shape and size. Without entering into the details, we simply refer to the results presented in (Blazek, 2005). For cell number i of volume V_i, we have the upper bound

$$\Delta t_i = \sigma \cdot k_i = \sigma \frac{V_i}{(\Lambda^{conv} + C_1 \Lambda^{cond})_i}. \tag{57}$$

The constants σ and C_1 depend on the spatial discretization, and should both be set to 2 for the first order upwind scheme we use. The spectral radii Λ are given as

$$\Lambda^{conv} = \sum_{j=1}^{m_i} \left(\left| \boldsymbol{u_{ij}} \cdot \boldsymbol{n_{ij}} \right| + c \right) A_{ij} \text{ and} \tag{58}$$

$$\Lambda^{cond} = \frac{1}{V_i} \sum_{j=1}^{m_i} b_{problem} A_{ij} \tag{59}$$

where $c = 0$ for incompressible flow and $b_{problem}$ is a constant depending on the problem properties, but not the geometry or grid. As diffusion plays a secondary role in this thesis, we refer to Blazek for any details on this restriction. We include diffusion only in the test cases presented in Section 4.3. Elsewhere, Λ^{cond} disappears as $b_{problem} = 0$, and the advective part reduces to

$$\Lambda^{conv} = \sum_{j=1}^{m_i} \left(\left| \boldsymbol{u_{ij}} \cdot \boldsymbol{n_{ij}} \right| \right) A_{ij}. \tag{60}$$

We shall compare the Equation 57 bound to the time steps used in Section 4, which of course all have been selected so that they respect the condition. However, remembering that the above condition is only an upper bound, we check that the time steps are not unnecessarily small. This is done by increasing the time step slightly and verifying that instability is observed. Because we use the time step as a criterion to rank the methods, such a precaution is prudent to ensure that no method is censored too severely.

To sum up the relevance of the CFL criterion to the results presented in this thesis, we repeat that the time step restriction is dictated by the smallest cell size to flow ratio of the domain. In cases of permeable fractures, the smallest ratio will be found within these almost without exception, and more often than not in the intersection cells, if any such are included in the model. Herein lies the primary motivation for ridding oneself of the intersection cells, either by the procedure described for the Final Volume method in Section 3.2 or by the scheme presented in Section 3.6. Furthermore, the local variation of time step bounds implies that with heterogeneous flow fields and grids, there will usually be parts of the domains where a considerably higher time step would suffice. In these regions, the small time steps lead to enhanced numerical diffusion compared to a simulation of the same region with a larger time step.

As an alternative to the explicit discretization of Equation 54, we discretize Equation 50 implicitly in time. By this, we mean that the new temperature value is determined from the value of the other terms also at the new time:

$$\frac{T_i^{n+1} - T_i^n}{\Delta t} + \sum_{j=1}^{m_i} \boldsymbol{u}_k T_{upw,ij}^{n+1} - D_{ij}\left(T_j^{n+1} - T_i^{n+1}\right) = f_i^{n+1} \text{ or} \tag{61}$$

$$T_i^{n+1} + \Delta t \left(\sum_{j=1}^{m_i} \boldsymbol{u}_k T_{upw,ij}^{n+1} - D_{ij}\left(T_j^{n+1} - T_i^{n+1}\right) - f_i^{n+1} \right) = T_i^n. \tag{62}$$

As is evident from Equation 62, this leads to a coupled system of equations to be solved, as new unknowns T^{n+1} of several different cells appear in each equation. The advantage is that this scheme is unconditionally stable in time, i.e., the solution does not blow up regardless of the time step. However, long time steps introduce artificial numerical diffusion which compromises the quality of the solution. The severity of this drawback is reduced if there is substantial physical diffusion in the problem, that is κ not too close to zero. This implies that the implicit discretization with rather large time steps is the method of choice for heat transport simulations

of moderate flow if the main concern is computational time. We have included a simulation series using implicit time discretization in Section 4.3 to explore its effect on the intersection cell elimination procedure presented in the next section.

3.6 Intersection Cell Elimination

As we will show in the following chapters, a significant error is introduced by the elimination of the intersection cells for certain classes of cases. Although not always obvious upon direct analysis of the pressure and flux fields, we encounter accumulations of error in the temperature field of considerable magnitude. Inspired by this, we want to investigate the possibility to remove the intersection cells after the flux simulation, but before the transport simulation. In other words, we have to find a way to determine the fluxes on the direct connections between the "normal" fracture cells meeting at the intersection from fluxes computed on the faces between each of these cells and the intersection cell. These new fluxes will then be used for evaluation of the advective term of Equations 54 and 62.

We begin by noting that for any of the involved non-removed cells c_i meeting at the intersection, the total flux over all the connections the cell takes part in must equal the flux $\boldsymbol{u}_i$ over the removed face between c_i and the intersection cell. If not, the mass conservation of c_i is violated. Denoting the new flux between cells i and j by $\boldsymbol{u}_{ij}$, we have one equation of the form

$$\boldsymbol{u}_i = \sum_{j=1}^{n} \boldsymbol{u}_{ij} \tag{63}$$

for $i = 1, \dots, n$, n being the number of fractures meeting at the intersection. We thus have n equations, which however only suffices for Y intersections where three fractures meet. For X junctions of four fractures, which have a total of six connections after removal, we need additional relationships.

To close the system, we have identified two approaches depending on the physics we wish to describe. The only difference between them is whether or not we assume that the transported quantity mixes in the intersection cells. In the full model, where the intersection cells are not removed, this mixing happens. To see this, consider that all the outfluxes from the intersection cell contain concentrations depending on all the influxes to that same cell. This behaviour should be in accordance with the local thermal equilibrium assumption. However, one could imagine relatively fast fracture flows with little diffusion. Then there might be too little time from the flow enters the tiny intersection cell till it leaves it again for mixing to occur. In this case, we might prefer to let the flow from a cell upstream of the intersection cell to the appropriate one downstream without adding information about the heat in the other upstream cell(s). The

differences show up in point 3 of the list of elimination procedures below. The non-mixing case is described in a and the mixing in b.

All flux directions in the following are as seen from the intersection, with fluxes out of the intersection cell denoted by positive values. We define the ratio between the number of influxes and the number of outfluxes as

$$R_{dir} = \frac{number\ of\ influxes}{number\ of\ outfluxes}. \tag{64}$$

We use assumptions of unidirectional, laminar flow in the fractures. The first entails that there can be no flow between two cells with fluxes of the same sign, i.e., two influx or two outflux cells. The latter takes slightly different form depending on the geometry of the flow.

1. Knowing that laminarity should imply that the flux from any given cell be unidirectional, we conclude that if $R_{dir} = {}^{1}/_{3}$, all fluxes must go from one cell and that all the other cells have influxes of magnitude equal to the magnitude of the pre-elimination fluxes into these cells. Thus, we have three non-zero fluxes, and three zero fluxes between the three "other" cells. The same is true with inverted flux directions, i.e., $R_{dir} = 3$.
2. For $R_{dir} = {}^{2}/_{2} = 1$ and influxes entering on opposing sides, both influxes must split according to the magnitude ratio between the outfluxes. This is because it is the only way we can have laminar flow where the fluxes merge to form the new outfluxes.
3. Lastly we have the scenario of a $R_{dir} = 1$ with the old influxes entering from neighbouring cells. This is where we have to choose between the above sketched approaches. If we want to model mixing flow, we use the following:
   a. We can be certain that one of the outfluxes is no larger than its neighbouring influx and must originate entirely from this neighbour. If not, the fluxes would have to cross, which is not possible in a laminar flow.
   b. If, on the other hand, we allow the heat time to mix in the intersection cell, the concentration in each of the outfluxes should be equal. This means we are again in the splitting regime 2 described above.

As we compare to a reference solution *with* intersection cells for the transport, we will focus our investigations on option 3 b. Option 3 a will be briefly explored in Section 4.2, but then abandoned because it models a slightly different physical problem than the other methods, i.e., one where the assumption of thermal equilibrium introduced in Section 2.4 does not apply.

In 2, our argument only considered the fluid's behaviour. Incidentally, this is equivalent to

weighting according to the 3 b argument about heat mixing, and so we have only one procedure to consider for case 2.

The physical explanation of our elimination procedure can be further substantiated by tracing the streamlines inside the intersection cell. For a given flux field, we can do this by means of Pollock's method. Assuming the flux field to be piece-wise linear in each coordinate direction leads to a semi-analytical solution (Thiele, 2001). By tracing the calculated streamlines from influx cells to outflux cells, we can confirm the fluid behaviour in all three cases.

For a schematic of the different flow patterns of a X intersection, see the left part of Figure 7, where we show the flux directions from each of the adjacent fracture cells. The thickness of the arrows is proportional to the flux over the corresponding boundary of the intersection cell streamline plots to the right. The latter are found by Pollock's method on a refined discretization of the intersection cell. At the top, we show the trivial $R_{dir} = {}^{1}/_{3}$, where the one influx splits into three outfluxes. Then the two cases of $R_{dir} = 1$ are shown, first the opposing case 2 and then the neighbouring case 3. The flux magnitudes listed in Table 1 are directly reflected in the number of streamlines leaving over each boundary face in the streamline plots. Most interestingly, the results show the splitting weighting in case 2. The more trivial behaviour, given that we can rely directly on unidirectionality and non-crossing flow in our model, of the other two is also confirmed.

	east	*north*	*west*	*south*
1 – one to three	*1*	*-4*	*2*	*1*
2 – opposing	*3*	*-4*	*2*	*-1*
3 - neighbouring	*3*	*2*	*-4*	*-1*

Table 1 lists the magnitude of the boundary fluxes used to compute the streamlines shown in Figure 7 by Pollock's method.

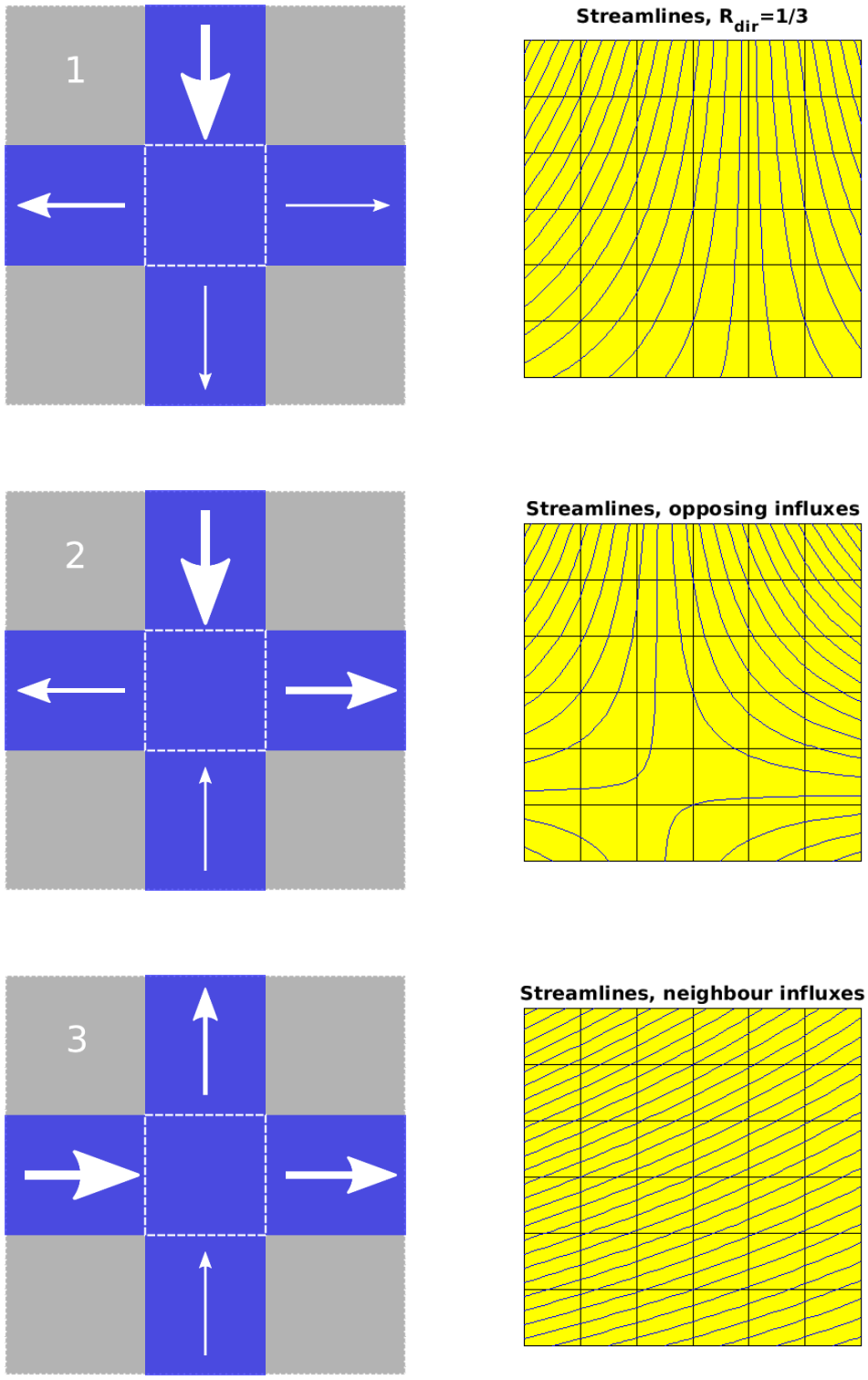


Figure 7: Three different intersection flux scenarios with corresponding intersection cell streamlines to the right.

In two dimensions, one usually settles for Y and X intersections. In three dimensions, however, one encounters junctions of six fractures even for very simple geometries. The approach with mixing is readily extendable to intersections of more than four fractures and can consequently be applied in three dimensions. The procedure would be the same as above: distinguishing

between in- and outfluxes and compute new fluxes for between influx cells and outflux cells according to the weights determined by the relative size of the original fluxes. The non-mixing approach may also possibly be extended, but more elaborate closure relations are necessary. One possibility would be to further exploit the streamline tracing.

Summing up, we have these the three options for flow simulations

1. Finite volumes with intersection cells
2. Finite volumes without intersection cells
3. Finite elements (here and always with intersection cells)

We can combine 1 and 3 with heat simulations with or without intersection cells. In case of the latter, we use the elimination procedure described in Section 3.6. As there is no reason to reinsert the intersection cells after the flow simulation but before the transport (this would only decrease the allowed time step, as explained in 3.5), option 2 is only run with transport simulations *without* intersection cells. All in all, we then have the five combinations shown in Table 2. We shall occasionally refer to the methods $\mathrm{FV_{ic}}$ and MFE_ic as the "full" methods and the $\mathrm{FV_{el}}$ and $\mathrm{MFE_{el}}$ as the "reduced".

	Flow	Transport intersection cells
$\mathrm{FV_{ic}}$	Finite Volumes with intersection cells	✓
$\mathrm{FV_{el}}$		
$\mathrm{FV_{no\ ic}}$	Finite Volumes without intersection cells	
$\mathrm{MFE_{ic}}$	Mixed finite elements	✓
$\mathrm{MFE_{el}}$		

Table 2 shows the five options of flow and transport discretization combinations investigated in this thesis.

3.7 Numerical Software Tools and Modifications

All grids used in the simulations presented in Section 4 are produced using the open-source meshing program Gmsh. We have used the 2d Delauney triangulation algorithm restricted by the specified fractures. The actual simulations were done using Matlab after we imported the triangulation coordinate list and connectivity matrix, as well as labelling vectors identifying the fractures and boundaries among the faces. The exception is the Section 4.4 MFE flow simulations, which were performed using Python.

The Finite Volume flow method is implemented as part of the Matlab Reservoir Simulation Toolbox (MRST), documented in (Lie, 2015). This is an open source toolbox of data structures and computational methods developed by the research institute SINTEF. The toolbox consists of a core module and several add-on modules. One of the latter is the Discrete Fracture Matrix module, developed by E. Keilegavlen and T. H. Sandve. They have extended MRST's grid structure to include fracture cell data and rewritten a number of core functions to handle fractures. Among these functions are the Finite Volume solvers both for two- and multipoint flux approximations and the functions computing the transmissibilities at the grid's faces. The streamline tracing by Pollock's method in Section 3.6 was also performed using the MRST implementation.

We use the Finite Element method as implemented by W. Boon. As this method is still being developed, the code is not yet publicly available. Two implementations exist, one in Matlab and one Python version. The latter uses FEniCS, a software package for solving differential equations.

The transport discretization is written using the grid structure of MRST. Furthermore, the heat diffusivities D_{ij} are computed by a call to the transmissibility calculation functions of the DFM module. In addition to this, we exploit the boundary condition structure and some minor functions. The advection term and upwinding are primitively coded from scratch. Once the fluxes are in place, the rest is just about setting parameters, assembling matrices and looping through the time steps.

We need the fluxes produced by the flow methods for the advective term. In the cases where the same MRST type grid is used for both flow and transport, namely $\mathrm{FV_{ic}}$ and $\mathrm{FV_{no\ ic}}$, the fluxes are available in a solution state structure. This structure has the same face (and cell) indexing as the grid structure, and it can therefore be directly imported for the flow simulations.

In the cases where we eliminate the intersection cells after the flow simulation, different grids are used for the two simulations. Let us call the grids G_{ic} and G, and the corresponding solution structures S_{ic} and S. We must then build a mapping from S_{ic} to S based on the geometry information in G_{ic} and G for the faces f_{ic} and f. There is also the case where the fluxes are computed with the MFE method. We apply a mapping from the geometry information as it is used in the Finite Element method, $\mathrm{G_{MFE}}$, and to a MRST structure with intersection cells. Pseudo code showing the workflow of the flow and transport simulations and the intermediate geometry mapping steps, see Figure 8. The rightmost column of Table 2 shows which of the four steps are relevant for each of the five method combinations.

Although we could have gone directly from the MFE structure to a G structure, we prefer the "detour" via G_{ic} to avoid recoding the somewhat cumbersome elimination step. The advantage

of this approach is that having coded the elimination once in the MRST framework, we can now in theory import the flux field from any method using intersection cells. As long as the fluxes and geometry data are importable to Matlab, we can write the mapping between geometries and eliminate the intersection cells using existing code.

As an example, we refer to Section 4.4 where the MFE flux field was computed using a Python implementation. This flexibility in flux field importing elimination with respect to software and programming language can be useful in at least two respects. Firstly, if one wishes to do more extensive comparisons involving more methods. And secondly to further investigate the consequences and possible benefits of eliminating the intersection cells between the simulations of flow and transport.

	Solution sequence
FV_{ic}	1,4
FV_{el}	1,4
$FV_{no\ ic}$	1,3,4
MFE_{ic}	1,2,4
MFE_{el}	1,2,3,4

Table 3: The simulation steps described in the pseudocode necessary for the different flow-transport combinations.

```
1)      Run flow simulation with FV or MFE to obtain fluxes.

2)      Build mapping G_MFE to G_ic:
        for all faces f_Gic
                Identify corresponding f_MFE by centroid coordinates.
                Identify flux direction by centroid coordinates of neighbouring cells.
        endfor
        Map fluxes from original solution structure to MRST S_ic.


3)      Build mapping from G_ic to G:
        for all new faces f_G not at intersections

                Identify corresponding old face f_Gic by centroid coordinates.
                Identify flux direction by centroid coordinates of neighbouring cells.

        endfor
        for all intersections
                Identify old faces f_Gic and new direct connections f_G at the intersection.
                Calculate new fluxes at the intersections using the Section 3.6 procedure.
        endfor
        Map fluxes from S_ic to S.


4)      Perform the transport simulation:
        Set initial temperature distribution
        if transport is solved with intersection cells
                Transport solver = f(S_ic, G_ic, rock properties, boundary conditions, wells)
        else
                Transport solver = f(S, G, rock properties, boundary conditions, wells)
        endif
        for all time steps
                Solve Equation 54 using Transport solver.
        endfor
```

Figure 8: Pseudocode describing the four main steps of the full flow and transport simulations.

3.8 Error Computation

When comparing results of numerical methods, one often seeks a way to quantify the quality of the results. In some cases, there is a reference solution available, i.e., either an analytical solution or a numerical solution which is much more precise than the results to be compared. When this is the case, one can measure the quality using the errors of the methods relative to the reference solution.

In general, the results to be compared may very well have been obtained using different spatial discretizations. This implies that one should compare only variables defined in the entire domain, such as pressure and heat. The fluxes would be a poor choice, as they are only defined on the faces of the different discretizations, which need not coincide.

The errors are in this thesis computed by taking the L^2 norm of the difference between fine and coarse pressure (or temperature) values on the fine grid for the matrix and fractures separately. This means that for any nonempty intersection of fine cell c_i and coarse cell C_j, we calculate the area of intersection a_{ij} and compare the pressure values. The error is the sum over all nonempty cell pairs normalized by division through the L^2 norm of the pressure range of the reference solution. This normalization value, which equals the size of the domain $|\Omega^k|$ times the magnitude of the pressure range, is chosen because the problem is driven by the pressure gradient, as seen from Equation 12. We thus arrive at the relative error

$$E_{rel,\Omega^k} = \frac{(p-P)_{L^2,\Omega^k_{fine}}}{(p_{max}-p_{min})_{L^2,\Omega^k_{fine}}} = \sqrt{\frac{\sum_{i,j} a_{ij}\left(p_i - P_j\right)^2}{|\Omega^k|\left(\max_{\Omega^k} p - \min_{\Omega^k} p\right)^2}}. \tag{65}$$

Here, Ω^k_{fine} denotes the matrix ($k = 2$) or fracture ($k = 1$) domain as partitioned in the reference solution, i.e., the fine grid. Where the error of the temperature field is evaluated, the form of Equation (65) is used, with temperatures t and T of the fine and coarse solution replacing p and P.

When we visualize the error locally, we wish to keep track of the sign of the error as well as its magnitude. Consequently, we use a direct difference rather than the local L^2 value in such plots. The error plots are of $\mathrm{T_c} - \mathrm{t_c}$, or coarse minus reference temperature, throughout the thesis. It is done for each fine cell c, so $\mathrm{T_c}$ is the temperature of the coarse cell which cell c is a refinement, or part, of. In some of the cases presented in the next chapter, there is no fine scale solution. We can still compare the coarse solutions cell by cell, so long as we take care to use the same mesh for all solutions. We shall refer to such comparisons as *differences* rather than errors, and take care to specify which solution was subtracted from which. Where applicable, we subtract the solution we place more confidence in from the one we consider more simplified.

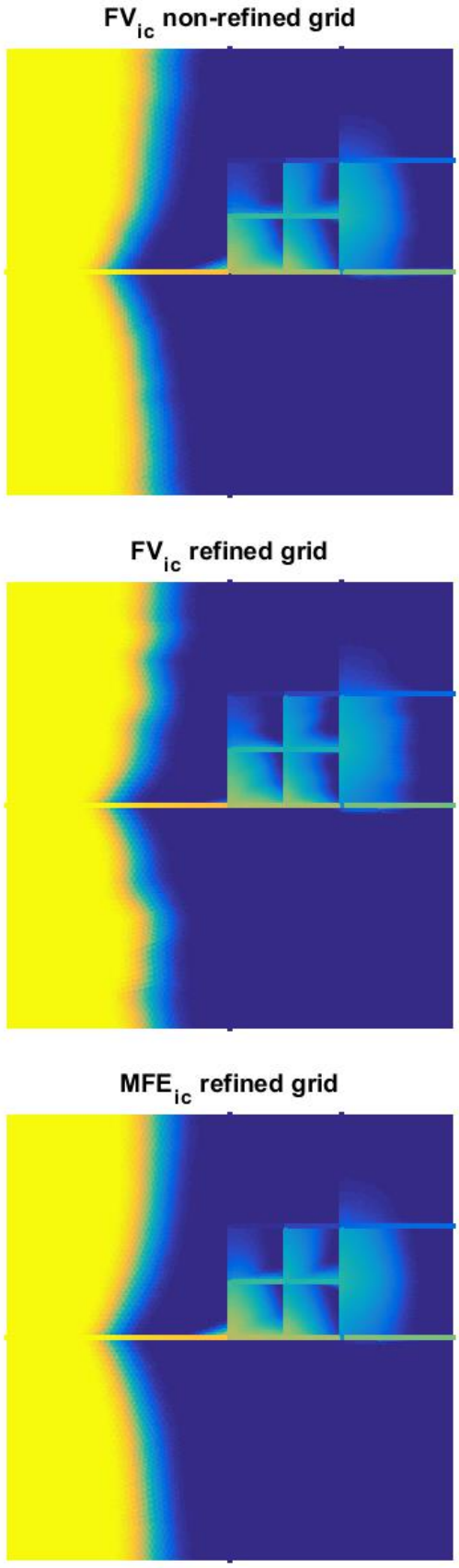


Figure 9: The temperature front of the FV_{ic} solution is irregular when the grid is a refinement of a coarser one. It is smoothened by using a normal unstructured grid. The behaviour is not observed for the MFE_{ic}.

As we consider the elimination of the intersection cells a simplification, we have not performed it in the computation of the reference solution, reducing our options to combinations $\mathrm{FV_{ic}}$ and $\mathrm{MFE_{ic}}$ of Table 2. To avoid information loss in the comparison plots between fine and coarse solutions, the grid used for the reference is a refinement of the grids used for the coarse simulations. When we perform transport simulations using FV flux fields on such grids, we observe some unhealthy grid orientation effects at the temperature front. We attribute this to the artificial structure inherent in grids which are results of successive structured refinements of much coarser ones. The problem is remedied by changing to a "normal" grid, i.e., a direct Delauney triangulation with approximately as many cells as the refined one. However, we choose to use $\mathrm{MFE_{ic}}$ for our reference solutions to be able to use refined grids. Without having investigated the matter thoroughly, we assume that the reason the problem is not observed for the $\mathrm{MFE_{ic}}$ is its relative abundance of degrees of freedom (pressures and fluxes as compared to only pressure in the FV). Examples illustrating the irregularity are shown in Figure 9. Note that the fractures are made artificially thick in the plots to make them visible. The same is done throughout Chapter 4.

3.9 Computational Cost

In addition to evaluating the quality of the results through error computation, one would also want to know at which computational cost they came. The most straightforward method of comparison is to compare the computation times of the different methods. But this would not always be fair or accurate, because the results presented in the next chapters were obtained on several different computers of varying and moderate processors. Therefore, we seek some readily quantifiable indicators to measure the computational cost. As described in Sections 3.2 and 3.3, the flow computation comes down to solving a linear system of the type $\mathbf{A}\boldsymbol{x} = \boldsymbol{b}$, where $\boldsymbol{x}$ are the discrete unknowns in each cell (pressure and, in the case of the MFE, fluxes), $\boldsymbol{b}$ incorporates the boundary conditions and sources and $\mathbf{A}$ describes how the unknowns are connected according to the spatial discretization we have applied. The cost of solving this system depends on the matrix $\boldsymbol{A}$, and we use the following indicators:

- The number of degrees of freedom, or unknowns, as it dictates the size of the solution matrix.
- The sparsity of the matrix, or the fraction of non-zero elements to the total number of entries of the matrix. A good numerical solver of linear systems can exploit the structure of a sparse matrix to greatly reduce the computational cost.
- The condition number, C. This does not strictly speaking influence the cost of solving the system. Rather, it is the factor by which the solution changes for a small change in the right hand side data. A high condition number means the error of the solution will be significant for even very slight errors in the data.

As explained in Section 3.5, the cost of performing a transport simulation very much depends on the time step restriction, so this will be our primary indicator. Indeed, we can use it as sole indicant, as the following considerations justify:

The number of degrees of freedom determines how many equations have to be solved. We will use the same mesh for each method choice in each test case. Thus, we will have the same number of degrees of freedom for the temperatures, bar the ones associated with the intersection cells, which might or might not be included. However, the number of intersections is in all cases too small compared to the total number of degrees of freedom for this to be of any consequence. Lastly, the spatial discretization of the transport equation determines the number of terms to be evaluated when solving the equations in each time step. This factor is also avoided, as we also use the same spatial discretization for all cases and methods.

4. Results

Using the two flow methods and different flow-transport combinations described in the previous chapter, we will study a selection of test cases for comparison of the methods. As part of a comparison project led by Bernd Flemisch, University of Stuttgart, the two methods have been tested on six different test cases for flow problems. Six different methods developed at the universities of Stuttgart, Milan and Bergen are compared in the work. The results are given in (Flemisch, et al., 2016), which is attached at the end of the thesis. We will use some of the test cases with slight modifications in this chapter, and refer to the appendix for the other scenarios. By showing examples computed on different test case geometries and changing parameters also for different runs on the same geometry, we aim to add weight to the conclusions we draw within the limitations of extrapolating from limited data sets.

We will start in Section 4.1 with a case where only the pressure fields are investigated and compared. This subsection is very similar to the results presented for the same test case in the comparison project paper. By also summing up the main comparison paper findings related to the two methods, we hope to establish their adequacy at solving pure pressure problems.

In Sections 4.2 through 4.4 we turn to cases where we also perform transport simulations. The six test cases are ordered according to complexity. In Section 4.2, both geometry complexity and the heterogeneity of the parameters are kept very moderate, enabling us to isolate the causes for the observed behaviour. In that section, we study the two different elimination procedures and point to a minor drawback of solving the flow without intersection cells. We take the same test case simplicity approach in the subsequent section, which only differs from the previous in that we change the permeability of some of the fractures.

We introduce the second test case geometry in Section 4.3, where effects of diffusion and implicit time discretization are examined. We do not aim at a thorough investigation on these points for the following reasons: Introducing diffusion rather obscures our main objectives, namely the insights about the quality of the flow methods and the properties of the intersection cell elimination procedure. Furthermore, a comprehensive analysis on the behaviour with both explicit and implicit discretization is beyond the scope of this thesis. Therefore, we merely suggest some general ideas about the influence of diffusion and temporal discretization on the importance of how the flow is solved.

We conclude the chapter with a test case of considerably higher geometrical complexity. By showing that the features identified in the preceding sections reappear in this case, we seek to establish the more general validity of the conclusions drawn from the other cases to more realistic applications.

4.1 Flow

We have chosen to start by presenting the third test case of the comparison project, which we will refer to as case a. The domain is the unit square, and all the strongly blocking fractures are parallel to one of the axes. We have both X and Y intersections, but no fractures ending inside the domain. The parameters of the case are listed in Table 4. Because of its geometrical simplicity and regularity, this test case is well suited to study the effect of isolated parameters, but we do not consider it to be a realistic subsurface fracture geometry.

In the case of the above mentioned comparison project, we have reference solutions computed on a very fine grid which all six evaluated methods are compared to. They were computed by Flemisch using either an equi-dimensional Cell Centred Finite Volume method with a multi point flux approximation, or a mimetic finite difference method. Both methods are implemented in the DuMu^{x} modelling package developed in Stuttgart, see Appendix I or (Flemisch, et al., 2011) for details. These reference solutions enable us to compute the error of each of the methods as described in Section 3.8, leading to an indication of the quality of each method's results.

The solutions are presented in pressure plots in Figure 10. The errors relative to Flemisch's reference solution and the parameters revealing the computational cost introduced in Section 3.9 are listed in Table 5. In addition to the FV without intersection cells and the MFE, which were the ones used in the comparison study, we also include the FV with intersection cells. From these results, we make the following observations:

- The errors of the three methods are very similar.
- The somewhat better result of the MFE method comes at the cost of a considerably higher number of degrees of freedom, although with a sparser matrix.
- The prize of the slightly smaller fracture error of the FV_{ic} compared to the $FV_{no\ ic}$ is a condition number orders of magnitude larger.

BCs	Top	$\boldsymbol{u} \cdot \boldsymbol{\nu} = 0$
	Bottom	$\boldsymbol{u} \cdot \boldsymbol{\nu} = 0$
	Left	$\boldsymbol{u} \cdot \boldsymbol{\nu} = -1$
	Right	$p = 1$
Fracture aperture		$a = 10^{-4}$
Matrix permeability		$K_m = 1$
Fracture permeability		$K_f = 10^{-4}$
Cells	With intersection cells	1490
	Without intersection cells	1481

Table 4: Parameters used for the case a pressure simulation. The following format is used to describe the test cases: one cell describing each of the boundary conditions, the permeabilities, apertures and the initial condition and final time where appropriate. Finally, we list the number of cells. The boundary flux values are positive for fluxes out of the domain. As all permeabilities and conductivities used are isotropic, the scalars K and κ replace ***K*** *and* ***κ****.*

Method	Matrix error	Fracture error	Non-zero fraction	Condition number	DOF
FV with intersection cells	$5.7 \cdot 10^{-3}$	$4.4 \cdot 10^{-3}$	$2.7 \cdot 10^{-3}$	$5.7 \cdot 10^{6}$	1490
FV without intersection cells	$5.7 \cdot 10^{-3}$	$4.6 \cdot 10^{-3}$	$2.7 \cdot 10^{-3}$	$2.6 \cdot 10^{4}$	1481
MFE	$4.5 \cdot 10^{-3}$	$4.9 \cdot 10^{-3}$	$1.6 \cdot 10^{-3}$	$1.2 \cdot 10^{5}$	3366

Table 5: Pressure errors and indicators of computational cost for case a.

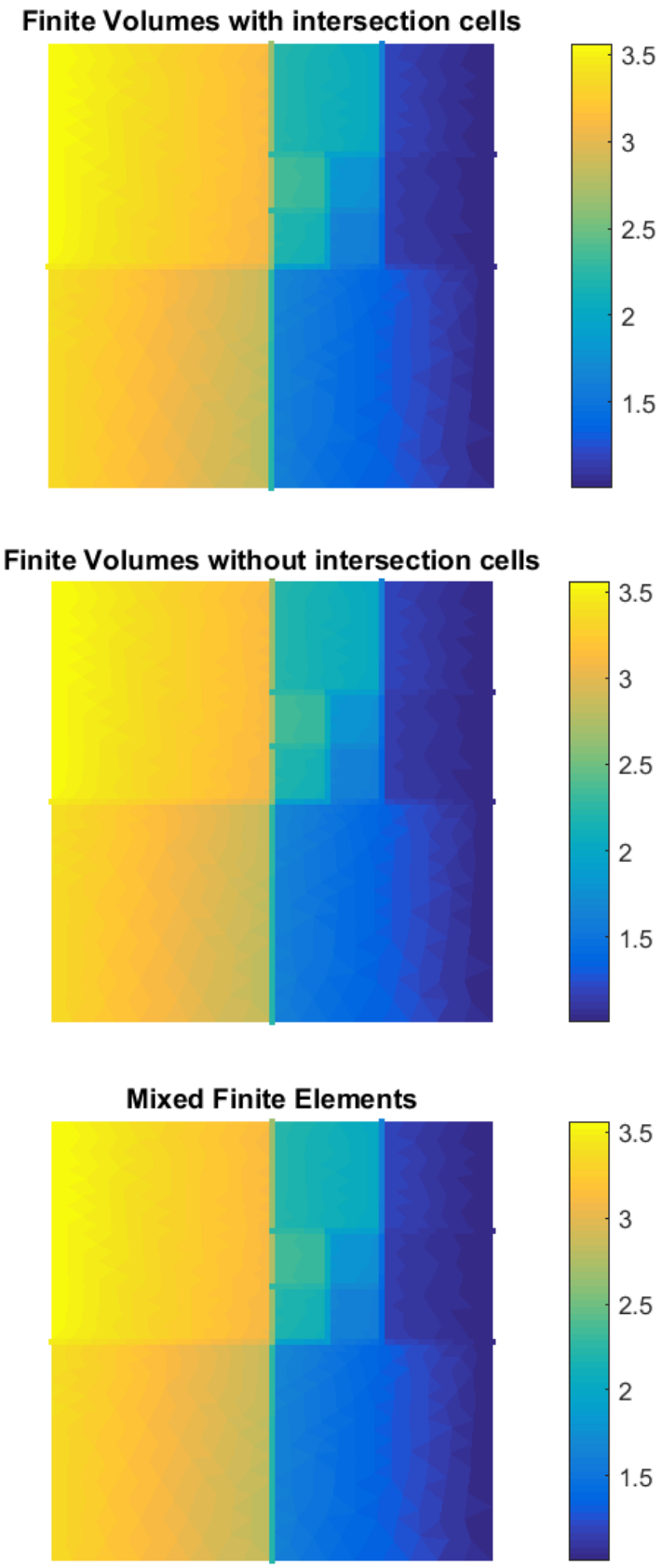


Figure 10: Pressure fields of the regular fracture test case a. The pressure drops significantly over the blocking fractures, especially the vertical ones which are normal to the pressure gradient and flow direction.

Both methods solve this blocking fracture case with quite satisfactory accuracy. The most striking observation is the huge improvement in condition number obtained by going from the FV with intersection cells to the one without. Apart from this, there is little to tell the three methods apart from these data. We are also satisfied by seeing that the fracture errors are not significantly larger than the matrix error for two reasons. Firstly, and most importantly, the argument for bothering to explicitly account for the fractures is that they will be important and perhaps dominant in determining the flow for the problems we are studying. Therefore, we would have been rather disappointed if our methods were to yield poorer results in the fractures. Secondly, a large fracture error might question the validity of the assumptions made in the fracture modelling, e.g. the co-dimension one approach and the removal of intersection cells.

Taking both fractures and the matrix into account, the methods of this thesis have of the smallest errors of all the methods of the comparison study for our test case a. In particular, they reproduce the pressure discontinuity over the vertical fractures. This is not achieved by all six methods. In fairness, some of the other methods display slightly more favourable errors compared to computational cost for the purely permeable fracture cases.

The main disadvantages of "our" two methods revealed by the comparison study are the following: The MFE condition number blows up in the purely permeable cases. It is also quite expensive in terms of degrees of freedom. Without going into the details, we would like to comment that there is an option of eliminating unknowns keeping the results unaltered. This leads to a denser matrix, which reduces the gain in terms of computational cost.[2] Due to the absence of intersection cells, the FV error increases considerably in the case where the fractures have different permeabilities (same case as presented in Section 4.3). We will further investigate and discuss this problem in Sections 4.2 through 4.4. Before we delve into this, however, we want to stress that the method does very well in the other pressure cases in the Appendix I comparison study, where the fractures are of homogenous permeability. With its low condition number and ability to capture discontinuities in the solution caused by blocking fractures, it emerges from that comparison as a good option if the intersection cell information is not crucial for capturing the overall flow. What is more, even these cases could be solved satisfactorily if one does not eliminate the intersection cells. This is indeed demonstrated in the paper in a test case slightly different from the one in Section 4.3. But the conclusions drawn in the comparison project based on the eliminated version condition numbers might not hold for the full version.

[2] The MFE has otherwise the sparsest matrices of all the methods of the comparison study.

4.2 Linear Transport

We now go one step further in what we will call case b. After producing the flux fields, we use them in a linear, advective transport simulation as described in Section 2.4. The choice of test case is made keeping in mind that our main interest is to investigate the validity of the assumptions we make in the fracture handling, in particular where the fractures intersect. Therefore, we again use the geometry of the previous section, but now with permeable fractures as presented as case two in the comparison study. With a fracture permeability four orders of magnitude higher than that of the matrix and the boundary conditions listed in Table 6, we have a case strongly dominated by the flow through the fractures, enabling us to isolate the behaviour as the tracer reaches the intersections.

As stated in Section 3.8, we use a refined $\mathrm{MFE_{ic}}$ as the reference. The temperature field comparisons are made at the final simulation time throughout Section 4. In Figure 12, we compare the reference temperature solution to coarse solutions produced by all combinations of Table 2. The temperature field errors listed in Table 7 are computed as explained in Section 3.8.

This is the only test case where we will use both the elimination procedures 3a (non-mixing) and 3b (mixing) described in Section 3.6, but only for the FV flux field. For the MFE flux field, we restrict ourselves to the full version $\mathrm{MFE_{ic}}$ and the $\mathrm{MFE_{el}}$ where the intersection cells are eliminated by the mixing procedure. All in all, we simulate temperature distributions from six different combinations.

As an extra justification for using the reference solution described, we compare the pressure field to the even finer and equi-dimensional reference solution of (Flemisch, et al., 2016) as described in the previous section. The matrix and fracture errors are $2.4 \cdot 10^{-3}$ and $1.2 \cdot 10^{-3}$ as compared to $1.9 \cdot 10^{-2}$ and $9.3 \cdot 10^{-3}$ for the coarse MFE solution. This implies that we have a satisfactory convergence for the pressure field, at least, and feel correspondingly more confident in our transport reference solution.

		Flow	Transport
BCs	Top	$\boldsymbol{u} \cdot \boldsymbol{\nu} = 0$	$\boldsymbol{J} \cdot \boldsymbol{\nu} = 0$
	Bottom	$\boldsymbol{u} \cdot \boldsymbol{\nu} = 0$	$\boldsymbol{J} \cdot \boldsymbol{\nu} = 0$
	Left	$\boldsymbol{u} \cdot \boldsymbol{\nu} = -1$	$T = 2$
	Right	$p = 1$	$T = 1$
Fracture aperture		$a = 10^{-4}$	
Fracture permeability		$K_f = 10^4$	
Matrix permeability		$K_m = 1$	
Fracture diffusivity		$\kappa_f = 0$	
Matrix diffusivity		$\kappa_m = 0$	
Initial temperature		$T_0 = 1$	
Simulation time		$t_{end} = 0.4$	
Cells	Coarse with intersection cells	421	
	Coarse without intersection cells	412	
	Reference	23801	

Table 6: Simulation parameters of test case b. The flow conditions are the same as for test case a, but the fracture permeability is drastically changed.

Method	Matrix error	Fracture error	Number of time steps
$\mathrm{FV}_{\mathrm{ic}}$	$1.063 \cdot 10^{-1}$	$4.81 \cdot 10^{-2}$	$1.6 \cdot 10^{7}$
$\mathrm{FV}_{\mathrm{el,\ mixing}}$	$1.063 \cdot 10^{-1}$	$4.81 \cdot 10^{-2}$	$1.2 \cdot 10^{4}$
$\mathrm{FV}_{\mathrm{el,\ no\ mixing}}$	$1.058 \cdot 10^{-1}$	$6.11 \cdot 10^{-2}$	$1.2 \cdot 10^{4}$
$\mathrm{FV}_{\mathrm{no\ ic}}$	$1.085 \cdot 10^{-1}$	$8.58 \cdot 10^{-2}$	$1.2 \cdot 10^{4}$
$\mathrm{MFE}_{\mathrm{ic}}$	$1.050 \cdot 10^{-1}$	$4.49 \cdot 10^{-2}$	$1.6 \cdot 10^{7}$
$\mathrm{MFE}_{\mathrm{el,\ mixing}}$	$1.050 \cdot 10^{-1}$	$4.49 \cdot 10^{-2}$	$1.2 \cdot 10^{4}$

Table 7: Relative errors and number of time steps of the different methods in test case b.

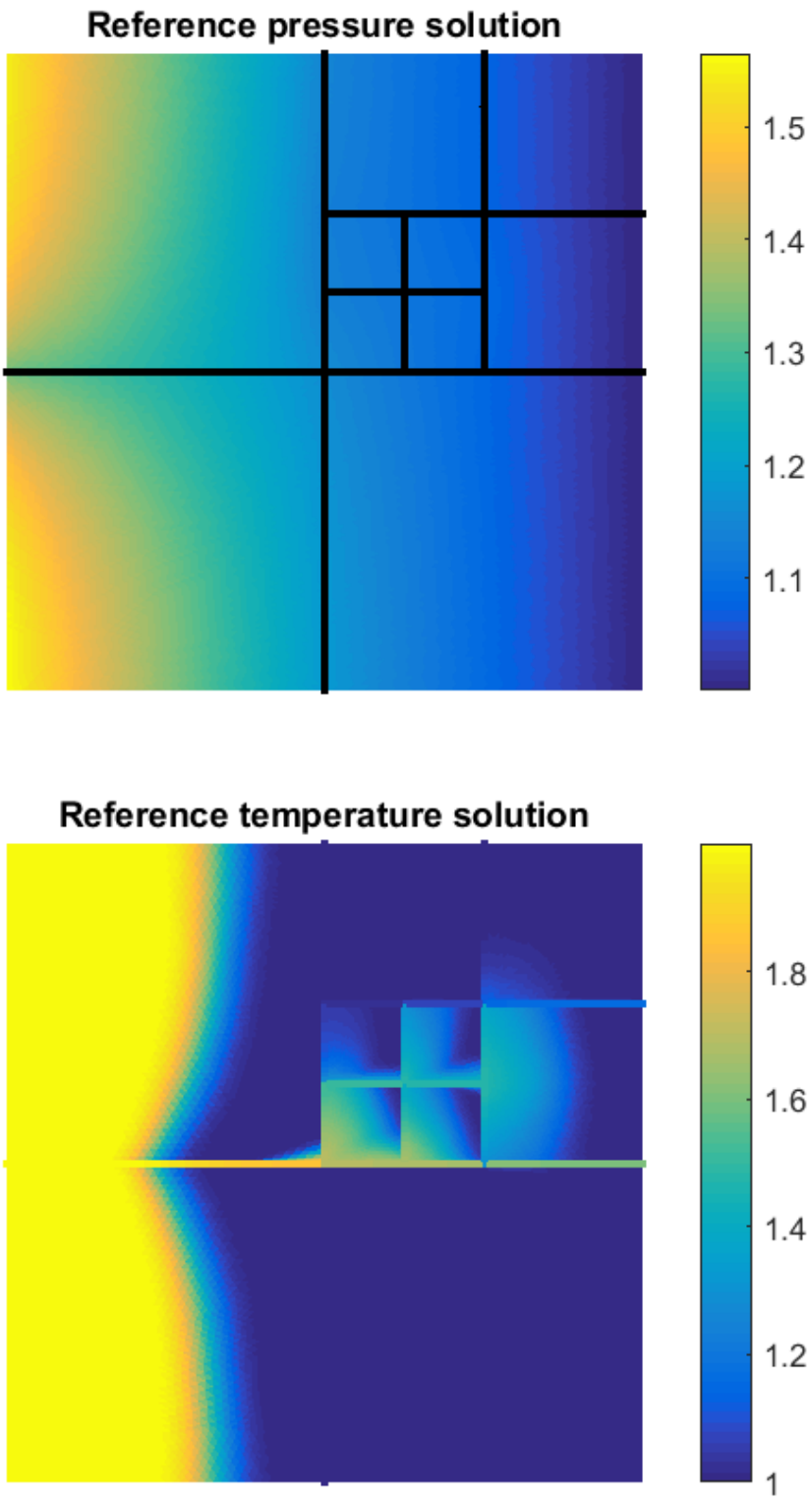


Figure 11: Test case b reference solution for the pressure and temperature distribution, the former with the fractures indicated by the black lines.

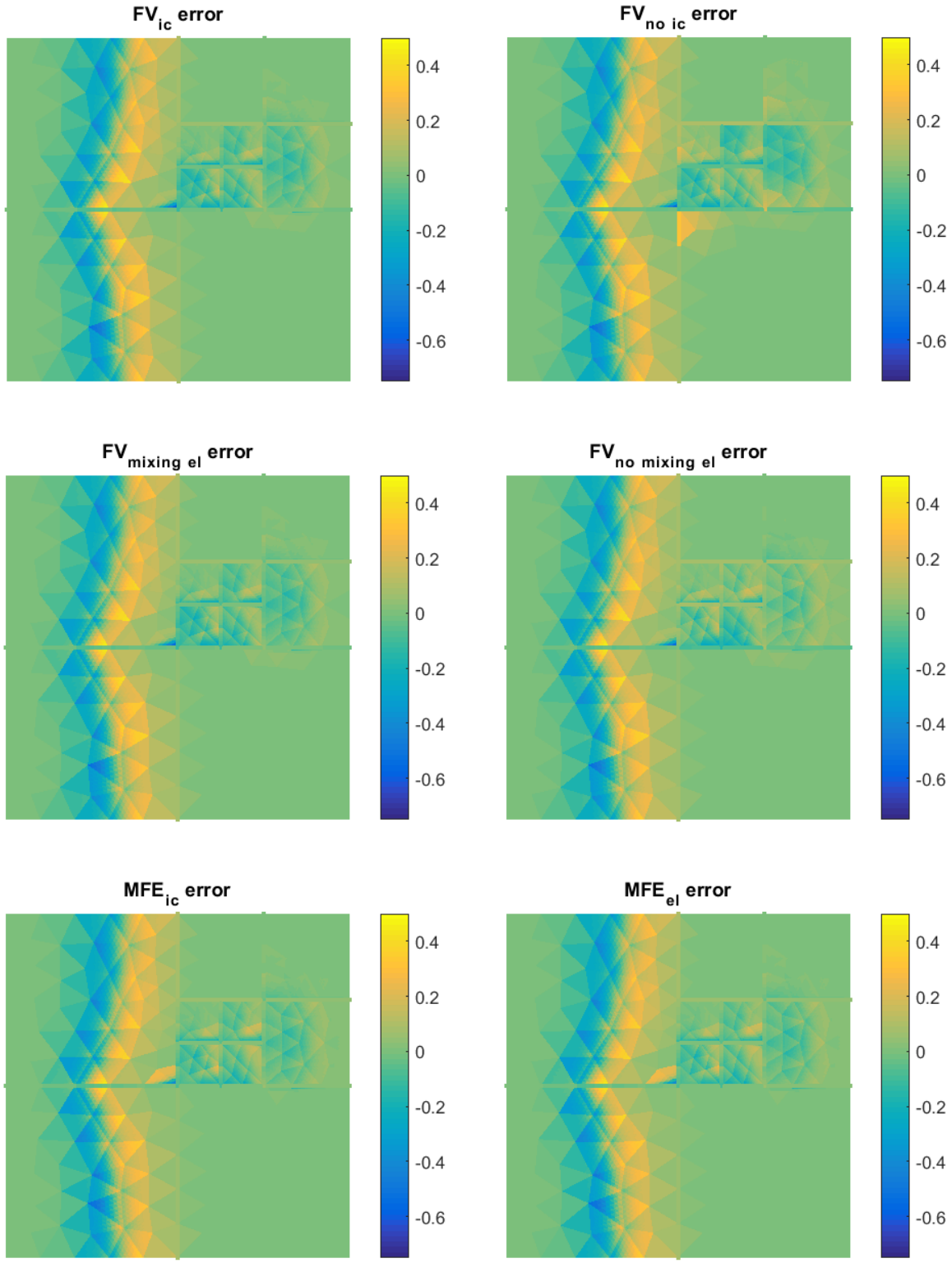


Figure 12: Case b error plots for all six combinations. The error at the temperature front dominates in all plots.

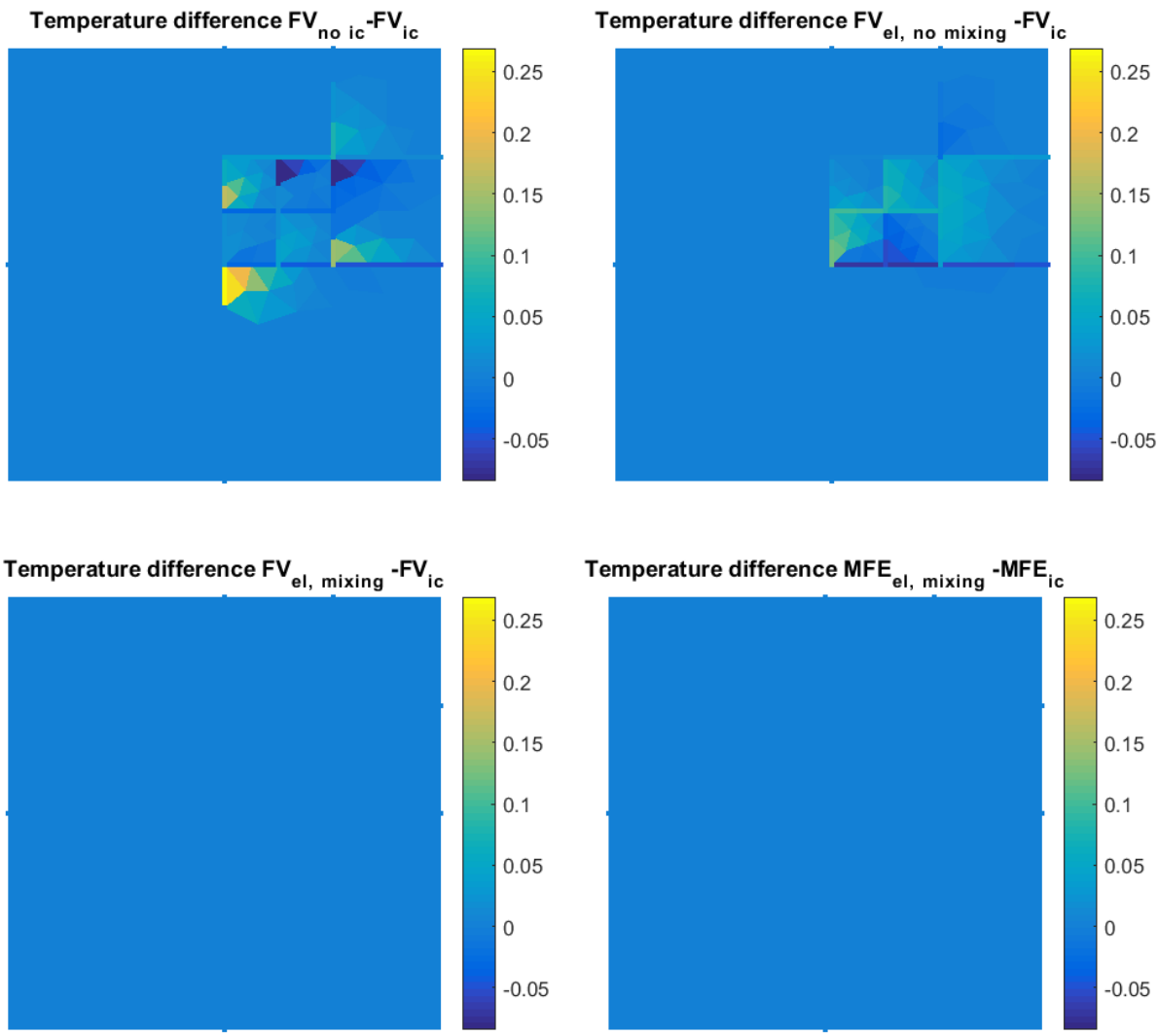


Figure 13: Cell-wise difference between the four combinations without intersection cells for the transport and the corresponding full combinations, all with the same colour axis. The differences in the two at the bottom are far too small to be detectable with this axis.

The errors listed in Table 7 and plots in Figure 12 and Figure 13 show the following:

- The two mixing combinations are virtually identical to the full ones, but come at a far lower computational cost.
- There are only minor differences between the two full combinations FV_{ic} and MFE_{ic}.
- The poorest result is obtained by the $FV_{no\ ic}$. The error plot shows that while the result is mostly comparable to the other ones, there are some additional local errors of considerable magnitude.
- The non-mixing combination solution is a bit different from the mixing one.

Before we set out to discuss these results in some more depth, we remind the reader that there will always be an error on the coarse meshes compared to finer meshes, in particular at the temperature front where the gradient is at its steepest. It seems to be somewhat above 10 % in these cases. Therefore, we try to see past this "background" error and compare the differences between the combinations in each case. This is the reason we include direct difference plots in Figure 13. We also repeat that the transport reference solutions were computed using the $\mathrm{MFE_{ic}}$, which might be considered unjust to the FV combinations. However, we observe good agreement between $\mathrm{FV_{ic}}$ and $\mathrm{MFE_{ic}}$.

The mixing elimination results are practically identical to those of their respective full combinations. No difference is found in any of the errors listed in Table 7, and the difference plot of Figure 13 is equally promising. Combining this with the time step information in Table 7, we deem this the most important observation of this subsection: Our computational cost improvement of approx. three orders of magnitude comes with no detectible loss of solution accuracy.

Even if we try to mentally subtract the background error, we cannot claim to demonstrate large differences between the errors of any of the six combinations. All differences between the values listed in Table 7 are more or less within the error variations to be expected on such coarse grids. The $\mathrm{FV_{no\ ic}}$ fracture error is the only one which might be considered significantly inferior to the others, but by a narrow margin. If we study the spatial distribution of the error, however, we make a rather more striking observation. The good part of the difference is located just downstream of the centre intersection. Similar behaviour can also be spotted at some of the other intersections, in particular the top and bottom of the rightmost three.

We explain the observation by considering that using the TPFA, a flux is computed between each of the four cells meeting at the intersection and all the three others. In the full method with intersection cells, no flow takes place between the two upstream cells (or downstream cells, respectively). Thus, artificial fluxes are produced, introducing hot water in a part of the domain where there should be none. The observation of this error was a strong motivation in the early stages of the development of the intersection cell elimination procedure. However, (Sandve, et al., 2012) showed that it can be avoided by choosing the multi point flux approximation. Even so, we appreciate that the results where we have used the Section 3.6 elimination procedure lie far closer to the full methods.

A discernible difference is found in the non-mixing $\mathrm{FV_{el}}$ combination, as we would expect from its representing a slightly different physical model than the other combinations. Not surprisingly, the plot in Figure 13 shows that the error is concentrated downstream of the intersection cells where the elimination differences come into play and the Table 7 values indicate that the impact

is strongest in the fractures. We will now abandon the non-mixing for the rest of this thesis with the remark that it is not a perfect substitute for the full method, but might indeed be considered in the physical context of where it belongs, i.e., when the local thermal equilibrium assumption is not valid.

We make an observation about local error for the MFE similar to the $\mathrm{FV}_{\mathrm{no\ ic}}$, albeit less pronounced. In this case the accumulation occurs immediately upstream of the intersections. The total magnitude of "leaked" heat decreases as the grid is refined for the reference solution in Figure 11, but the leakage is still observed and much larger than in the corresponding FV simulations. Placing rather more trust in the more thoroughly tested FV, we are not at the present sure whether this is a problem inherent to the MFE method or a mere bug in the code. From the point of view of a comparison study, it is worth mentioning that both these accumulation errors (MFE and $\mathrm{FV}_{\mathrm{no\ ic}}$) can serve as examples of errors which would not be noticed by an evaluation of the pressures.[3] Rather, they are revealed by the transport simulation, as was one of the main motivations of this thesis.

Because of the geometry regularity and high permeability of the fractures, this test case provides an opportunity to compare the time steps to the theoretical upper bounds from Section 3.5 without too much effort. As the fracture permeability is four orders of magnitude higher than that of the matrix, we may assume the highest flow rates to be found inside the fractures. Since the fractures are orthogonal to the axis and each fracture cell is a rectangle with two faces of non-zero projection on each axis, the projections reduce to

$$\Lambda^{conv} = A_{vertical}(|u_{east}| + |u_{west}|) + A_{horizontal}(|v_{north}| + |v_{south}|),$$

where $A_{vertical}$ and $A_{horizontal}$ the area of the vertical and horizontal faces, respectively. The u specific discharges originate from the *x*-direction fluxes over the vertical faces and the v specific discharges are the two over the horizontal faces, see Figure 14. This means that each component of the Λ sums is a total discharge.

Using the MFE coarse solution as an example, the smallest local criterion is $1.5 \cdot 10^{-8}$ and corresponds to the intersection cell at (0.75, 0.5). After the intersection cells have been eliminated, the smallest local value $1.9 \cdot 10^{-5}$ is found for the cell at (0.53, 0.5). Note that we here apply the criterion for cells with split intersection fluxes, and the result should be interpreted with a certain amount of caution. Keeping in mind that these should be conservative estimates, both values compare fairly well to the critical time step found by trial and error, namely $2.5 \cdot 10^{-8}$ and $3.3 \cdot 10^{-5}$.

[3] For example, the case a pressure errors of the $\mathrm{FV}_{\mathrm{ic}}$ and $\mathrm{FV}_{\mathrm{el}}$ are practically identical.

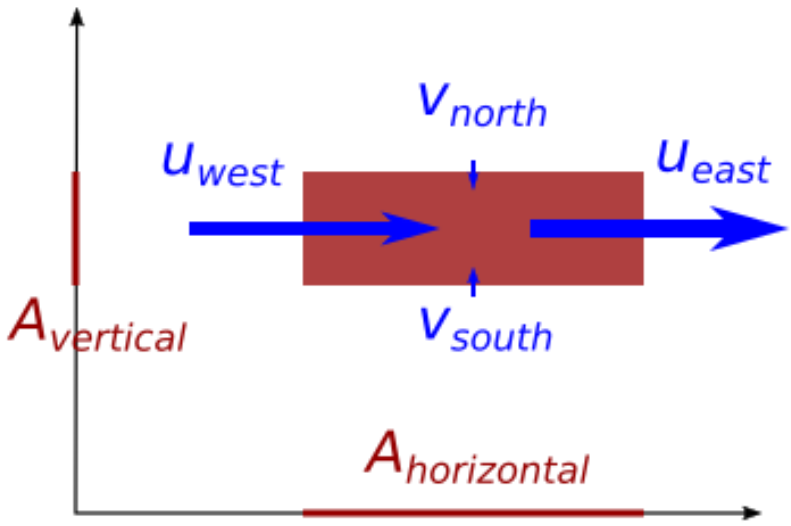


Figure 14: The fluxes and areas of a fracture cell such as in the time step criterion computation for the MFE_{el}.

We now proceed to define test case c. Keeping the geometry and boundary conditions of the previous cases, we now increase complexity slightly by letting the vertical fractures be blocking. Two series of simulations are produced for all combinations of Table 2, and compared to a refined $\mathrm{MFE_{ic}}$ reference solution. For each step in each of the series, we decrease the permeability of the vertical fractures. We characterize the cases by the fracture permeability ratio

$$r = \frac{K_{blocking\ fractures}}{K_{permeable\ fractures}} \tag{66}$$

which ranges from 1 to 10^8. In the first series, the intersection cells have permeability equal to that of the impermeable fractures, in the second one they are permeable. The former series mimics a geological history with the permeable fractures forming before the blocking, and vice versa in the latter.

By also changing the simulation time slightly (see Table 8), we try to reduce the risk of reporting results based on a special, nonrepresentative test case.

We plot the errors and the allowed time steps as a function of r in Figures 15 and 16. The fracture error for permeable intersections was omitted, as it behaves exactly as the matrix error and therefore does not provide additional information. In Figure 17, we show some example plots of the temperature distributions at the end of the simulations. These last plots are from simulations on significantly finer grids than those used in the simulation series. The purpose of this is to identify the errors which do not vanish as we refine the grid and to give an even clearer picture of the remaining qualitative differences. Finally, we monitor the temperature throughout the

simulation at a given location for two r values in Figure 18. We observe that:

- The differences between FV and MFE are again very moderate.
- Both elimination cases are virtually indistinguishable from their respective full combinations in the error plots.
- The $\mathrm{FV}_{\mathrm{no\ ic}}$ error explodes for high permeability ratios.

That both full methods display quite similar results is perhaps seen the clearest from the qualitative plot comparison in Figure 17. The marginal differences imply that there is again little to choose between them, but demonstrates how both handle the rather high permeability ratios of the fractures (up to $r = 10^8$). Remember that the Finite Volume demands non-refined grids, cf. Section 3.8.

Less tracer is seen in the matrix around the right part of this same fracture for the MFE than the FV in the qualitative plots. To investigate this and the slight differences observed in the relative errors, we examine the temperature in the rightmost $y = 0.5$ fracture cell for $r = 10^2$ and $r = 10^4$. The Figure 18 plots reveal a slight discrepancy. For the lower permeability ratio, they suggest that the tracer is transported slightly faster in the MFE simulations. While none of the data suffice for conclusions about significant differences, we will return to the point in Section 4.4, where we find similar differences.

The errors of the eliminated combinations are indistinguishable from that of the full ones for all values of *r*. Thus, the elimination procedure seems just as attractive where the fractures differ in permeability as it did in the previous section.

In the blocking intersection series we see the first example where a method is shown to be clearly inferior. The error plots in Figure 15 and temperature distributions shown in Figure 17 univocally demonstrate that the $\mathrm{FV}_{\mathrm{no\ ic}}$ produces completely false solutions for high permeability ratios. From the plotted fine grid temperature distribution, we see that the error is not that of a numerical scheme which has not yet converged due to too few degrees of freedom, but that it converges to a completely wrong solution qualitatively entirely different from that of the other combinations. This problem is *not* solved by applying the MPFA, for the simple reason that the information about the permeability of the intersection cell is entirely removed from the no-intersection model, regardless of the flux approximation chosen.

Fracture permeability	$K_f = 10^{-4} - 10^4$
Simulation time	$t_{end} = 0.6$

Table 8: The case c parameters which differ from the ones used in case b.

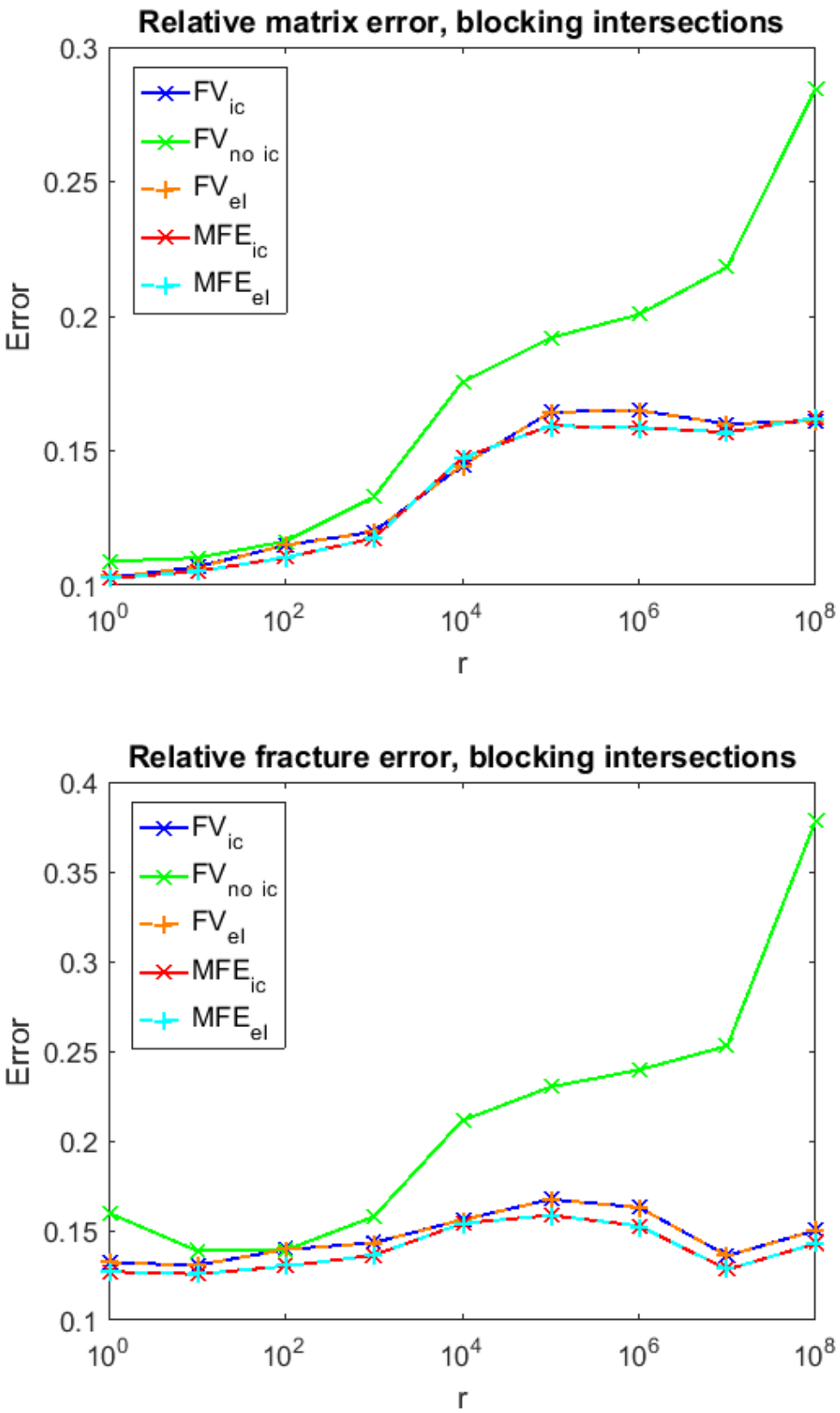


Figure 15: Relative errors for the test case c simulation series with impermeable intersections.

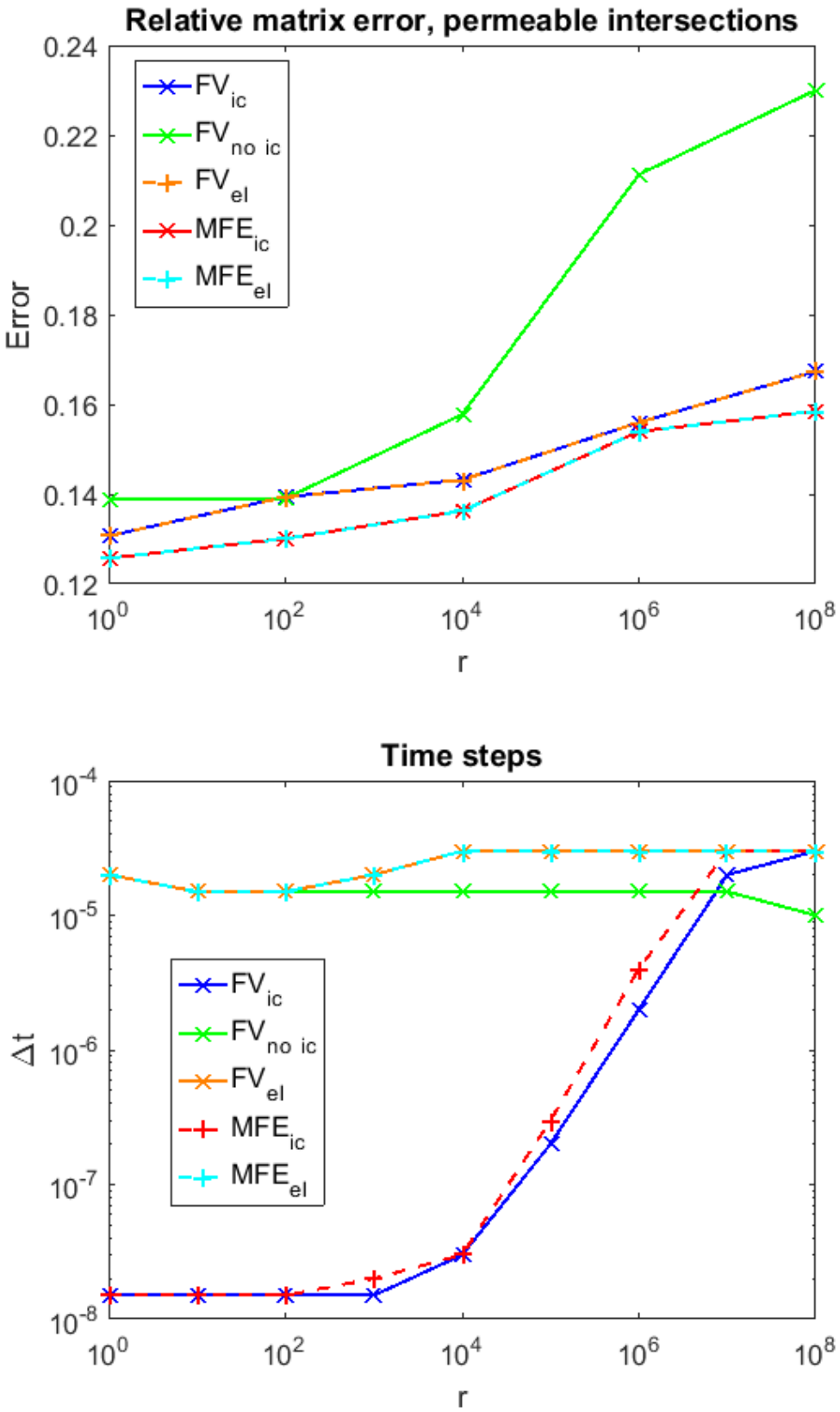


Figure 16: Relative error of the permeable intersection series and the time steps of the blocking intersection series.

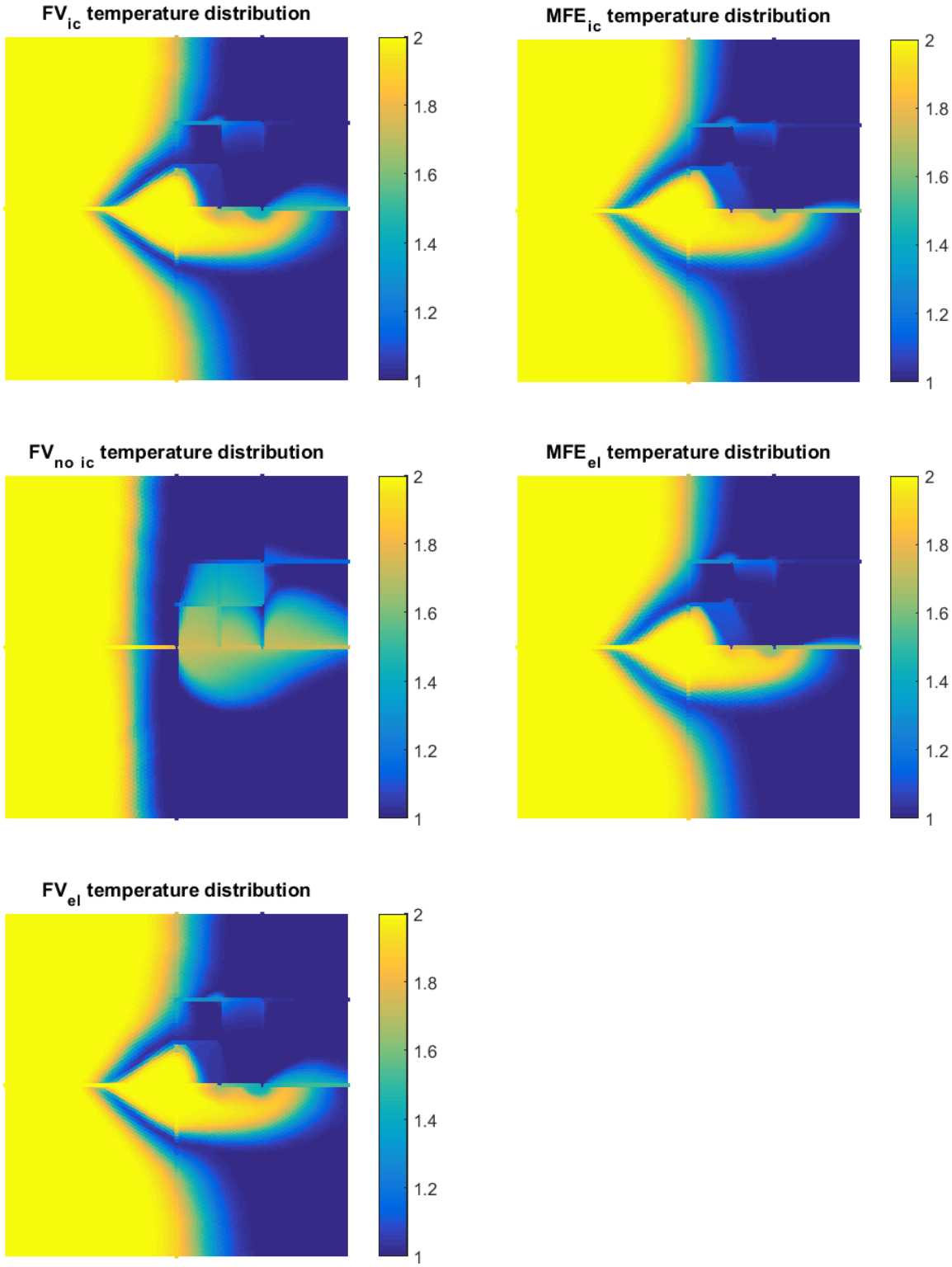


Figure 17: Refined solutions of the most extreme test case c permeability ratio, $r = 10^8$, for impermeable intersections. FV solutions are shown to the left and MFE to the right.

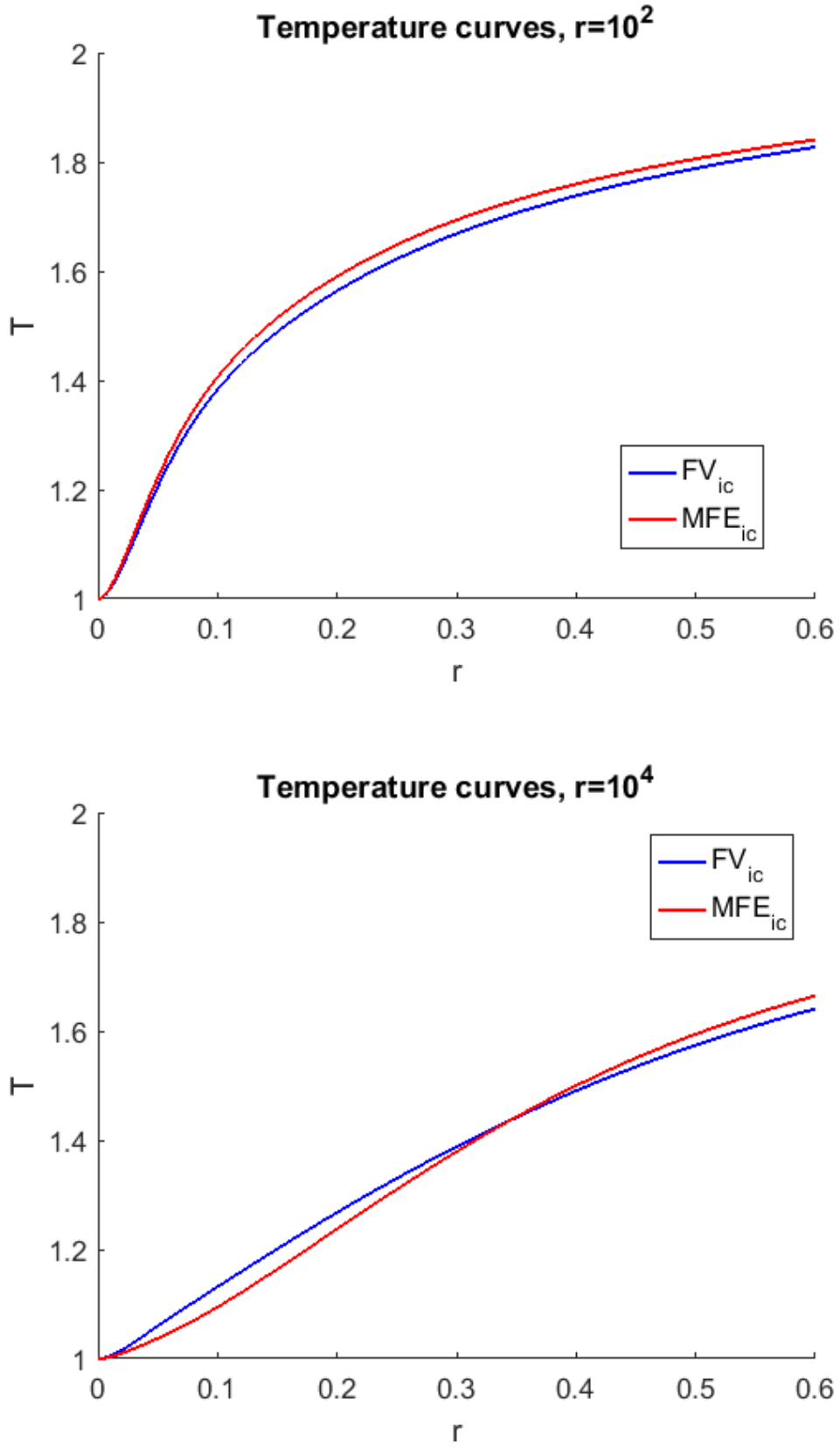


Figure 18: Temperature for the rightmost cell of the $y = 0.5$ fracture for two different r values.

There is an intermediate range of fracture permeability ratios of $\sim 10^1 - 10^3$ where the $\mathrm{FV}_{\mathrm{no\ ic}}$ does quite well, as seen in Figure 15. We assume this is because the blocking fractures no longer contribute to the flow so the artificial fluxes discussed above vanish, but the intersections, which are three to four orders of magnitude shorter than the fractures, do not yet block the flow along the permeable fractures significantly. Thus, the problem is almost equivalent to one with only the horizontal fractures present. Not until an even lower blocking fracture permeability leads to considerably reduced flow along the permeable fractures does the $\mathrm{FV}_{\mathrm{no\ ic}}$ break down.

In the case of the permeable intersection series, the inferiority of the $\mathrm{FV}_{\mathrm{no\ ic}}$ is much less marked. This is in good accordance with the idea of a blocking intersection being more decisive for the flow than a permeable one, cf. Section 2.5.

The idea is to use the elimination procedure in the cases where we warn against the $\mathrm{FV}_{\mathrm{no\ ic}}$. Our main argument for preferring the eliminated over the full combinations is that we relax time step restrictions by removing the intermediate cells. Now, we see from the time steps in Figure 16 that the time step difference between the full and the eliminated decreases as we increase *r*. We assume, based on the description of CFL criterions on unstructured grids in Section 3.5, that the smallest cells are the critical ones for permeable fractures. But by lowering the intersection permeability, we reduce the flow there. Thus, the local intersection cell CFL criterion is increasing in *r*, and at some point the smallest local CFL criterion is no longer at the intersections, but probably somewhere else in the fractures. At this point ($r \sim 10^6$, from Figure 16), we gain nothing in terms of time step restriction by removing the intersection cells.

The condition for the importance of the intersection CFL criterions to vanish, is that all intersections are blocking, as they are in this test case. If some are permeable, we cannot expect the flow to be (significantly) reduced there[4], and the global CFL will be much improved by eliminating the intersection cells also for high values of *r*, as we shall see in Section 4.4.

4.3 Heat Transport

We now want to investigate the effect of heat diffusion on the differences seen between combinations $\mathrm{FV}_{\mathrm{ic}}$ and $\mathrm{FV}_{\mathrm{no\ ic}}$ and the agreement between $\mathrm{FV}_{\mathrm{ic}}$ and $\mathrm{FV}_{\mathrm{el}}$. We choose a new geometry from (Flemisch, et al., 2016) for our test case d. It is also defined on a unit square domain and includes X intersections of different angles, fractures ending inside the domain and

[4] This depends on the geometry and actual fracture permeabilities. For example, a fracture with two intersections could be blocked by one of them, decreasing the flow all along the fracture and to the other, possible permeable, intersection. In this case, the local CFL criterion of both intersections *would* be lowered. However, in a real, complex geometry of some interconnectivity, we expect the blocking of some intersections to *increase* the flow over the other intersections of the connected pore network, as long as the connectivity is not lost altogether.

a situation where two fracture elements meet at an angle (see bottom right corner of the fracture distribution shown in Figure 21). Importantly, we again have fractures of different permeability meeting at some of the intersections. The permeability assigned to two of these three cells is the harmonic average of the permeabilities K_{f1} and K_{f2} of the crossing fractures

$$K_{intersection\ 1,2} = \frac{2K_{f1}K_{f2}}{K_{f1} + K_{f2}} = \frac{2 \cdot 10^4 \cdot 10^{-4}}{10^4 + 10^{-4}} = 2 \cdot 9.99999 \cdot 10^{-5} \approx 2 \cdot 10^{-4}. \qquad (67)$$

This means that the intersections are in practice impermeable, as in the second series of the previous section. The third intersection is permeable, i.e., $K_{intersection\ 3} = 10^4$.

We present three cases of the same geometry, boundary and initial conditions (Table 9), but with different, non-zero values of the diffusion coefficient. The reference pressure and temperature distributions are shown in Figure 19. For each of the three κ values, we perform one simulation with each of the three FV combinations. We compare the results in error plots in Figure 21 and the time steps and errors in Figures 20 and 21.

		Flow	Transport
BCs	Top	$p = 1$	$T = 1$
	Bottom	$p = 0$	$T = 0$
	Left	$\boldsymbol{u} \cdot \boldsymbol{\nu} = 0$	$\boldsymbol{J} \cdot \boldsymbol{\nu} = 0$
	Right	$\boldsymbol{u} \cdot \boldsymbol{\nu} = 0$	$\boldsymbol{J} \cdot \boldsymbol{\nu} = 0$
Fracture aperture		$a = 10^{-4}$	
Fracture permeability		$K_f = 10^4$	
Matrix permeability		$K_m = 1$	
Fracture diffusivity		$\kappa_f = 10^{-5}, 10^{-3}$ and 10^{-1}	
Matrix diffusivity		$\kappa_m = 10^{-5}, 10^{-3}$ and 10^{-1}	
Initial temperature		$T_0 = 0.2$	
Simulation time		$t_{end} = 0.3$	
Cells	Coarse with intersection cells	795	
	Coarse without intersection cells	790	
	Reference	11757	

Table 9: Parameters used for test case d.

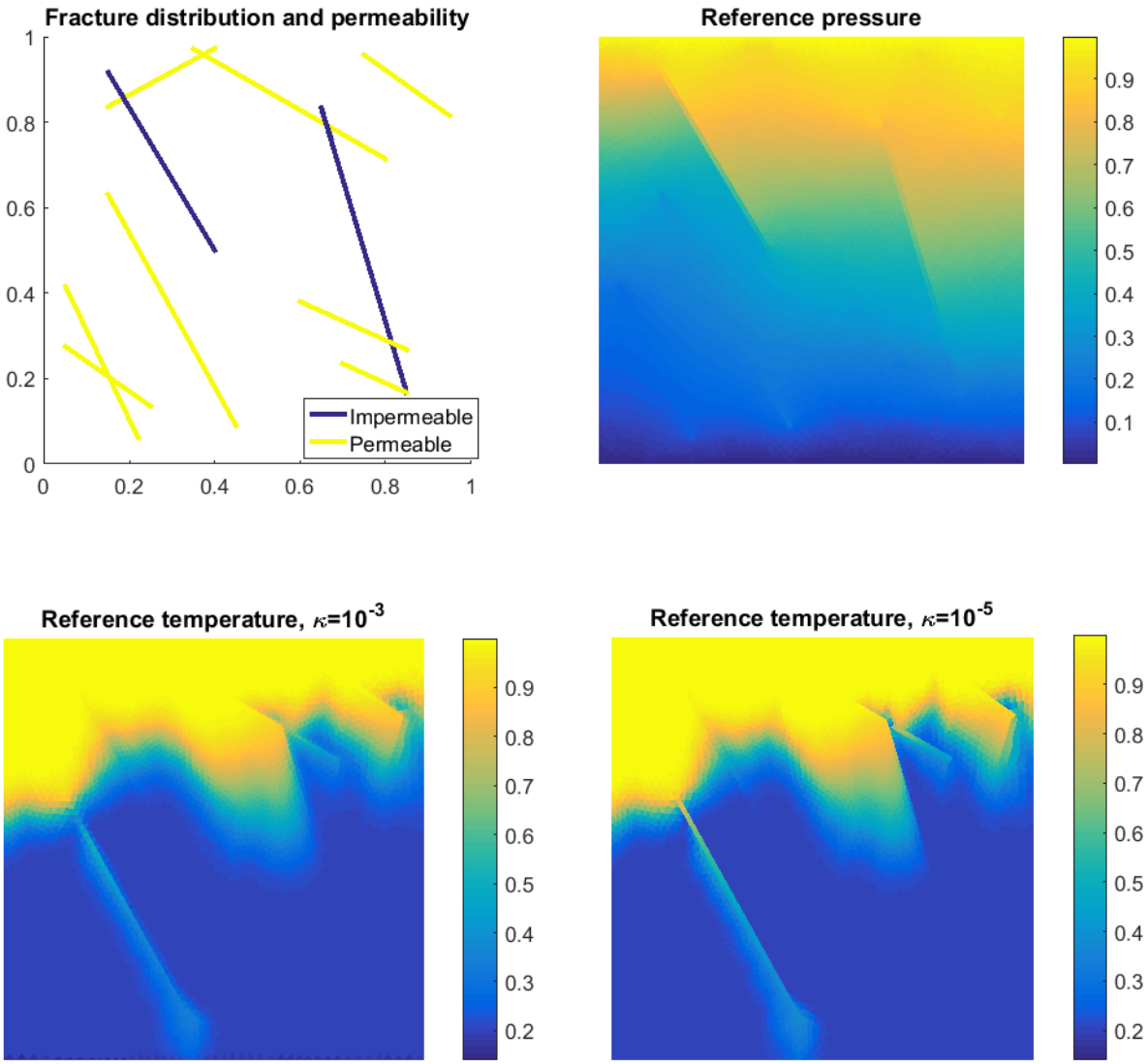


Figure 19: Fracture distribution and the reference solutions of the diffusion test case d.

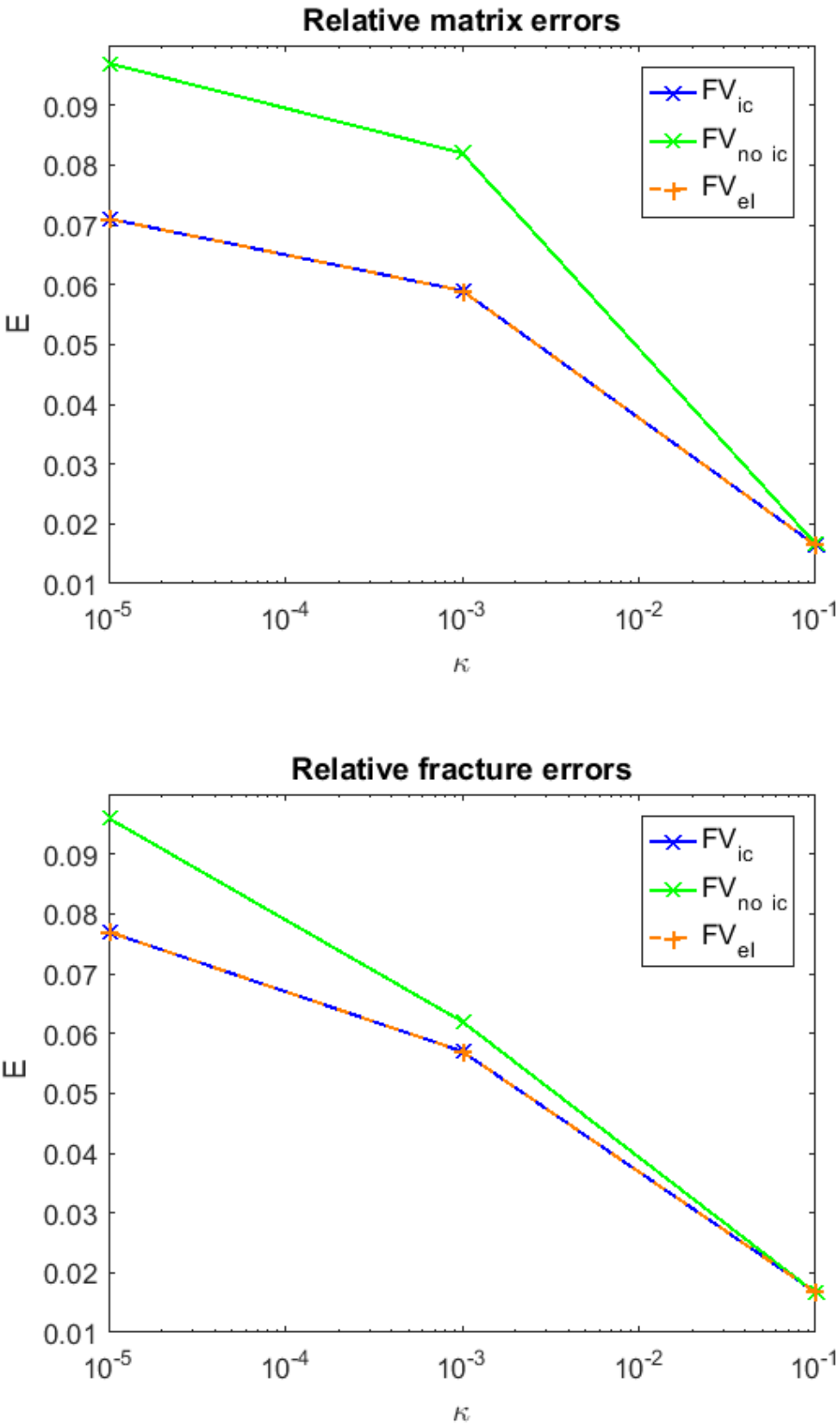


Figure 20: Relative errors for case d.

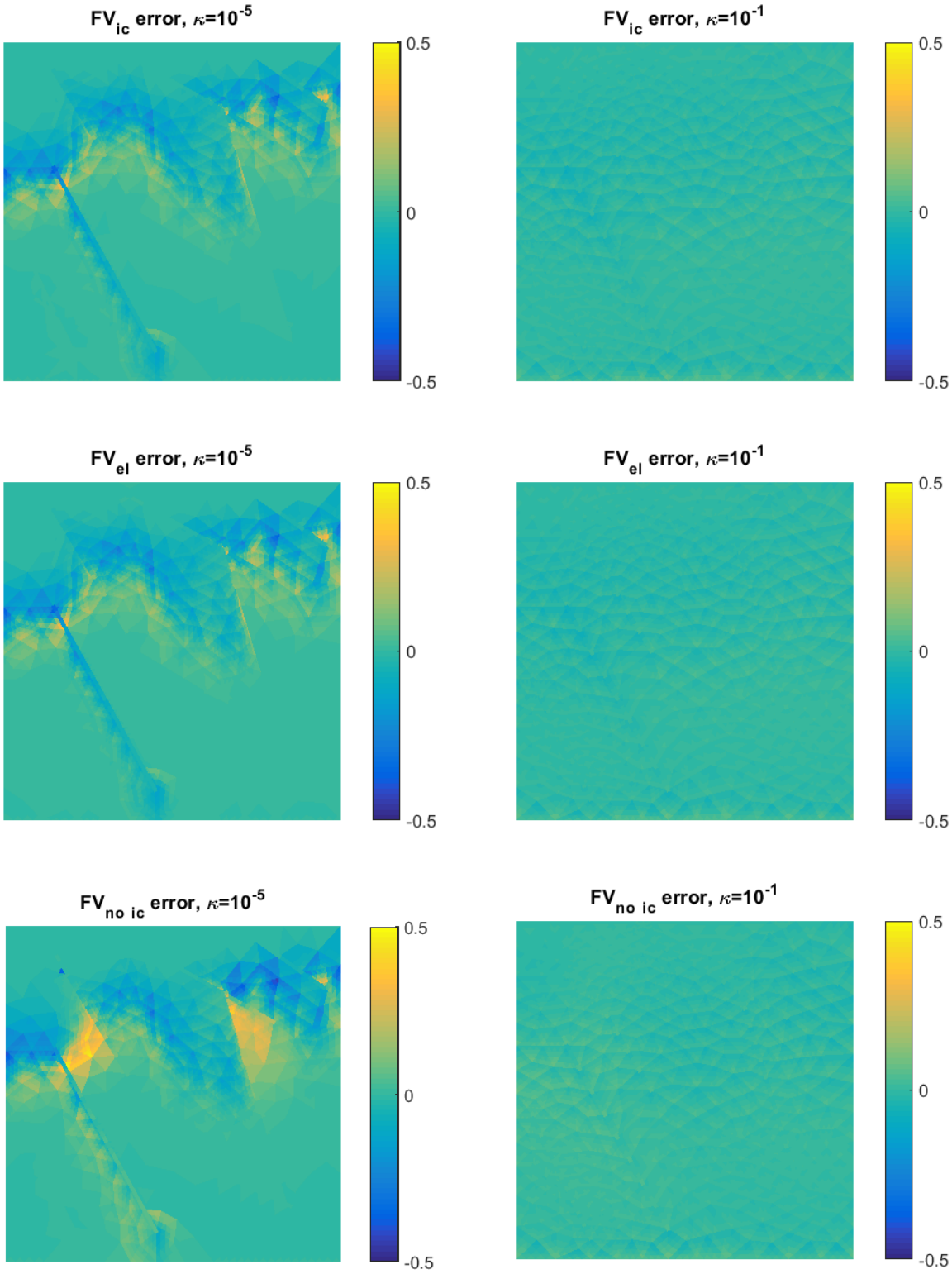


Figure 21: Error plots of the test case d simulations with $\kappa = 10^{-5}$ *to the left and* $\kappa = 10^{-1}$ *to the right.*

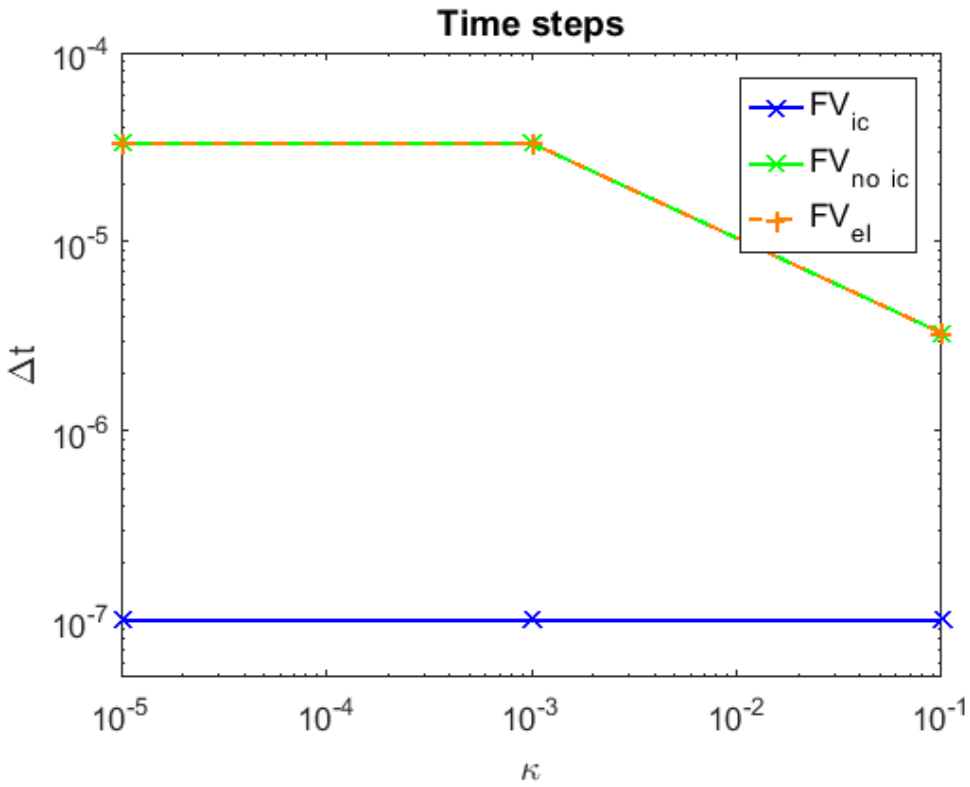


Figure 22: Time steps of the case d simulations for different diffusion constants.

The idea of this case is to investigate how the differences described and discussed in the previous sections, specifically between $\mathrm{FV}_{\mathrm{no\ ic}}$ and the other combinations, are influenced by diffusion. We note that

- The $\mathrm{FV}_{\mathrm{no\ ic}}$ error is considerably higher than the other for the two lowest κ values, but
- this inferiority decreases in κ, and is negligible for the diffusion dominated simulation.
- The eliminated results are just as close to the full ones as in previous test cases.

The results show that there are considerable differences between the full method and $\mathrm{FV}_{\mathrm{no\ ic}}$ not only in pure advective cases, but also when moderate diffusion is included. The Figure 21 $\kappa = 10^{-5}$ error plot shows that the good part of this additional error is located downstream of the blocking intersections, as one might expect.

The results of Figure 20 indicate that increasing the diffusion coefficient reduces the importance of how the flow was solved, and the differences in the heat distribution of the two combinations diminish accordingly. In the simulation where advection dominates, we see in particular one cell next to an intersection in the top left corner where the $\mathrm{FV}_{\mathrm{no\ ic}}$ combination produces a temperature completely different from the $\mathrm{FV}_{\mathrm{ic}}$ one. These kind of local errors are smoothed out

by diffusion. What remains are some more disseminated differences. There is, in other words, less reason to be worried about removing the intersection cells if heat diffusion dominates.

We yet again demonstrate near perfect agreement between the full and eliminated combinations. Based on this and the shortcomings of the $\mathrm{FV}_{\mathrm{no\ ic}}$, we suggest that the post-flow elimination procedure is relevant also in cases of moderate diffusion contribution, but that it can be replaced by the $\mathrm{FV}_{\mathrm{no\ ic}}$ in highly diffusive cases if desired.

For the rest of this subsection, we will turn our attention to a simulation series with implicit temporal discretization. For this test case e, we perform simulations with a series of diffusion coefficients ranging from zero to 10^{-1} with *implicit* time discretization and time steps of $10^{-5} - 10^{-3}$. The results relating to the condition numbers of the heat equation solution matrix are presented in Figure 24 and Figure 23. We find that

- Condition numbers are considerably improved by eliminating the intersection cells.
- The improvement decreases for larger time steps, but is significant in the time step range.

First, we refer to Figure 23 for the error of implicit runs compared to the explicit reference solution. For both values of κ, the error is quite similar to the explicit one for the time step range from 10^{-5} to 10^{-3}. For $\Delta t = 10^{-2}$, however, the error from the numerical diffusion is more pronounced, cf. Section 3.5. Therefore, we restrict ourselves to $\Delta t \in [10^{-5}, 10^{-3}]$, although one occasionally accepts some additional error in implicit runs to obtain results fast enough.

Comparing the condition numbers in Figure 24, we see that although the differences are not as striking as for the time steps in explicit runs, significant differences appear. We introduce the condition number ratio

$$R_c = {C(FV_{ic})}/{C(FV_{el})} \tag{68}$$

as an indication on the improvement in condition number resulting from eliminating the intersection cells. From the second plot in Figure 24, we observe that even if R_c becomes slightly smaller as we lengthen the time steps, $R_C > 10^2$ is observed for all but the highest κ value. We saw from Figure 23 that even the largest time step $\Delta t = 10^{-3}$ yeilds quite satisfactory results, meaning that one would prefer that value.

While these results suggest that something may be gained by eliminating the intersection cells also for implicit simulations, we also note that for the $\mathrm{FV}_{\mathrm{el}}$, the time step gain going from explicit to implicit simulations of comparable error quality is only between one and two orders of

magnitude, and so the computational cost improvement from the reduced number of time steps might be outweighed by the additional cost of solving the implicit discretization equation system, rendering the implicit option less tempting. This consideration depends on the particular parameters of the case one wishes to simulate and the accuracy one demands. Note also that all of the condition numbers in Figure 24 are too small to pose problems in this particular test case, but that they may increase for more complex scenarios.

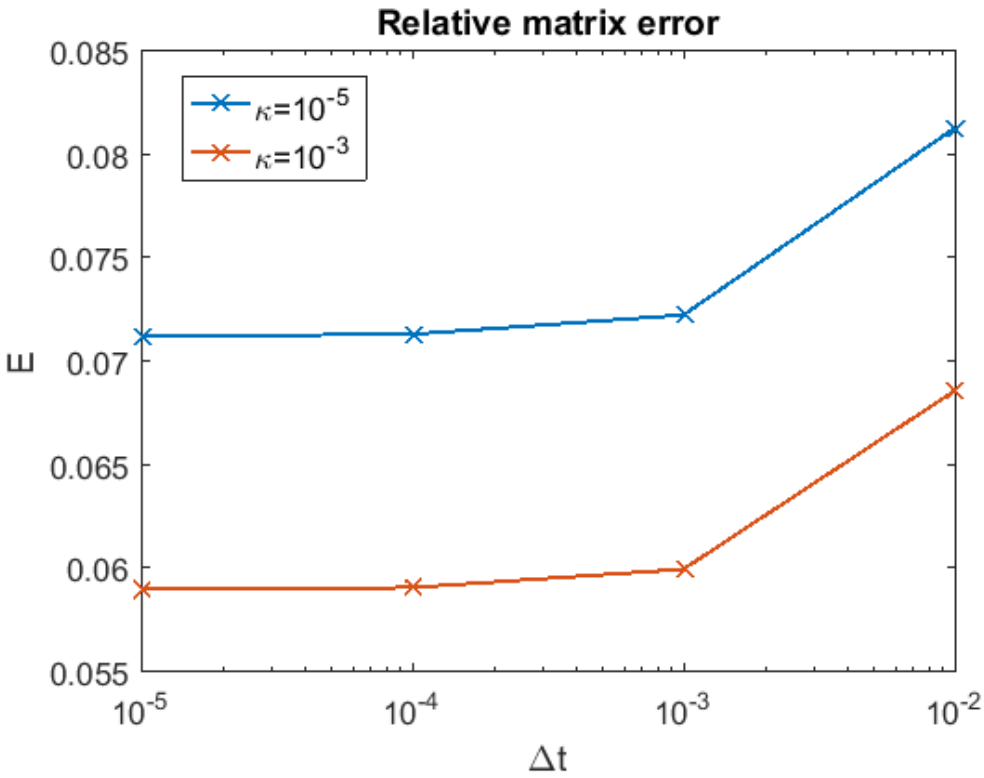


Figure 23: Relative matrix error of some of the implicit runs for different time step lengths.

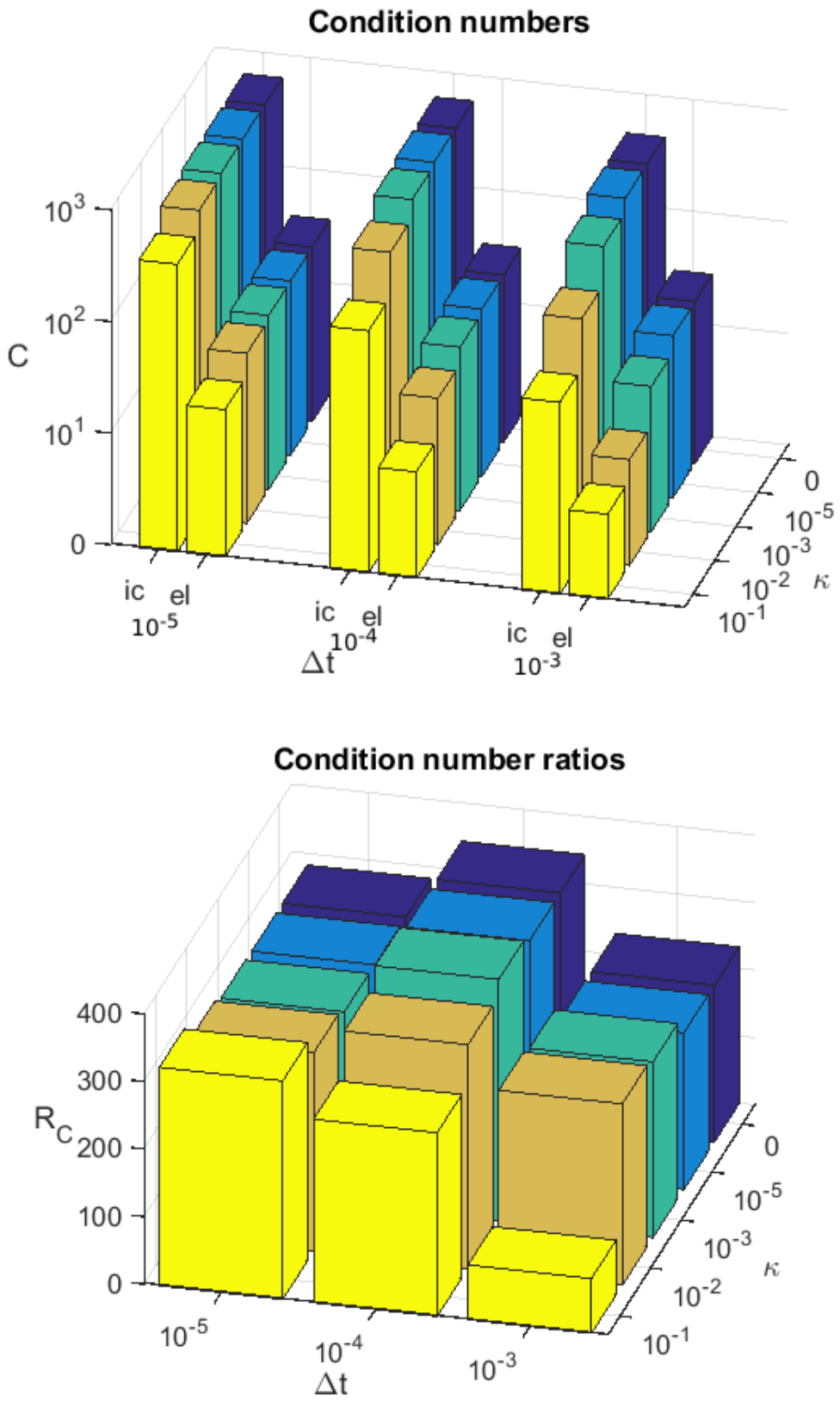


Figure 24: Condition numbers C plotted against the diffusion coefficient κ and time step Δt for the implicit simulations with and without the elimination at the top. At the bottom, we show how the condition number ratio R_c varies in time step and κ.

4.4 Complex Fracture Geometry

We now turn to test case f of more complex and realistic geometry to investigate the effects of larger networks of fractures. The geometry was obtained from satellite images of the island Sotra, situated approx. 15 km to the west of Bergen. Topological features seen at the surface were interpreted to reflect the subsurface fracture distribution by Luisa F. Zuluaga.

Because of the complexity of the geometry, the spatial discretizations require a certain minimum of cells to maintain grid quality, as explained in Section 3.1. This, in turn, means that we are not able to compute reference solutions with the full methods sufficiently more accurate than the coarse ones. However, we can do one refinement (four times as many cells) for the eliminated combinations. As these are not the full methods, we hesitate to regard them reference solutions. But due to the perfect fits shown between the full methods and the eliminated throughout the thesis, we do think they can provide information about convergence. Therefore, we include them in some of the comparisons but do not compute errors.

		Flow	Transport
BCs	Top	$\boldsymbol{u} \cdot \boldsymbol{\nu} = 0$	$\boldsymbol{J} \cdot \boldsymbol{\nu} = 0$
	Bottom	$\boldsymbol{u} \cdot \boldsymbol{\nu} = 0$	$\boldsymbol{J} \cdot \boldsymbol{\nu} = 0$
	Left	$\boldsymbol{u} \cdot \boldsymbol{\nu} = 0$	$\boldsymbol{J} \cdot \boldsymbol{\nu} = 0$
	Right	$\boldsymbol{u} \cdot \boldsymbol{\nu} = 0$	$\boldsymbol{J} \cdot \boldsymbol{\nu} = 0$
Wells	Injection	$\boldsymbol{u} = 1$	$\boldsymbol{T} = 1$
	Production	$p = 0.5$	$\boldsymbol{J} = -1$
Fracture permeability		$K_f = 10^{-8} - 10^{-16}$	
Matrix permeability		$K_m = 10^{-14}$	
Fracture conductivity		$\kappa_f = 1$	
Matrix conductivity		$\kappa_m = 1$	
Initial temperature		$T_0 = 2$	
Simulation time		$t_{end} = 10^4$	
Cells	$\mathrm{MFE_{ic}}$	9868	
	$\mathrm{MFE_{el}}$	9783	
	$\mathrm{FV_{ic}}$	15043	
	$\mathrm{FV_{el}}$	14957	
	Refined $\mathrm{FV_{el}}$	57380	

Table 10 lists the parameters of test case f. Well flux values are positive where fluid or heat is added to the domain.

We compare the coarse results directly. Using the same grids for all combinations of each method, we can compare the cells one by one in our analysis. We can also compare values locally. We set up a scenario with two wells. Mimicking and simplifying a geothermal scenario, we have one well injecting water of fixed, low temperature at a constant rate, and one production well with fixed pressure. Monitoring the temperature at the latter, we obtain information on differences in the flux field implicitly through differences in temperature and in particular the breakthrough time. We define the breakthrough time as the time when the first of the injected cold water reaches the production well, or more precisely the first time step when the temperature in the production well cell goes below 1.999.

In this final example, we have included all five method combinations. Again, we have performed a series of simulations with different permeability ratios. Whereas the spatial distribution of the fractures was determined by examining real geological data, the permeability assignment is done randomly. We assign permeabilities for the fractures with probabilities $P(permeable) = 0.25$ and $P(impermeable) = 0.75$. All the intersections are blocking, and the range of the ratio between conducting and blocking fractures is again from 1 to 10^8. In the parameter list of Table 10, the well parameters describe one well injecting fluid of constant temperature at a constant mass rate and one producing at a constant pressure, extracting fluid at the temperature of its location at any given time.

The temperature fields, difference plots, breakthrough time and final production well temperature plots in Figures 25 through 31 show similarities and differences between the combinations. We do the following observations:

- No restrictive problems arise from the complex geometry.
- Once again, $\mathrm{FV_{ic}}$ and $\mathrm{MFE_{ic}}$ are in fairly good agreement, but some differences are seen for intermediate r values.
- The elimination combinations are remarkably close to the full combinations.
- For the higher permeability contrasts, $\mathrm{FV_{no\ ic}}$ breaks down yet again, but
- at lower values of r, between 1 and 100, only very slight differences to the other combinations are observed.

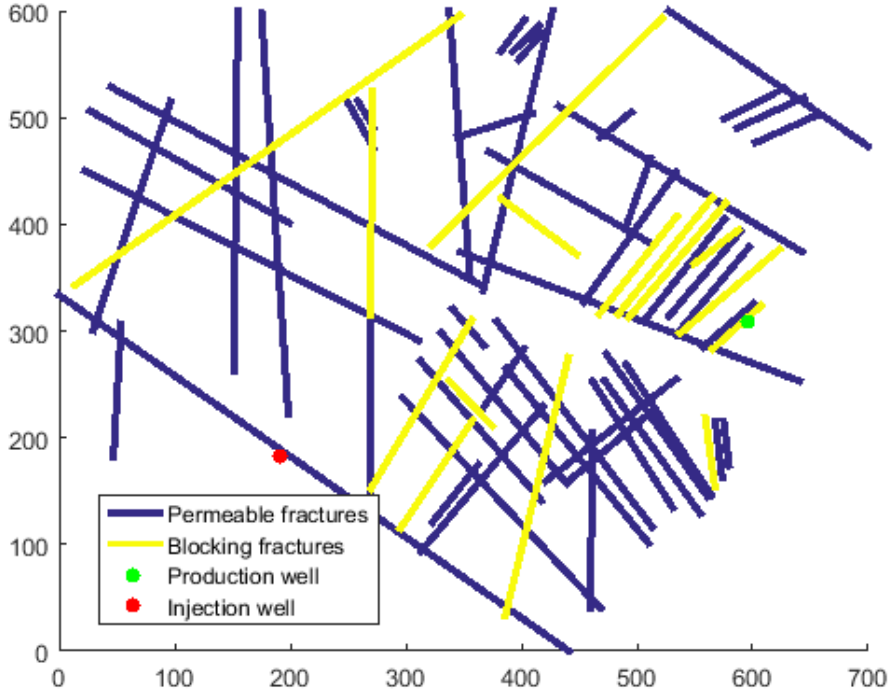


Figure 25: Fracture distribution and permeabilities and well placement for case f. Note that the wells are located next to, not inside, the fractures.

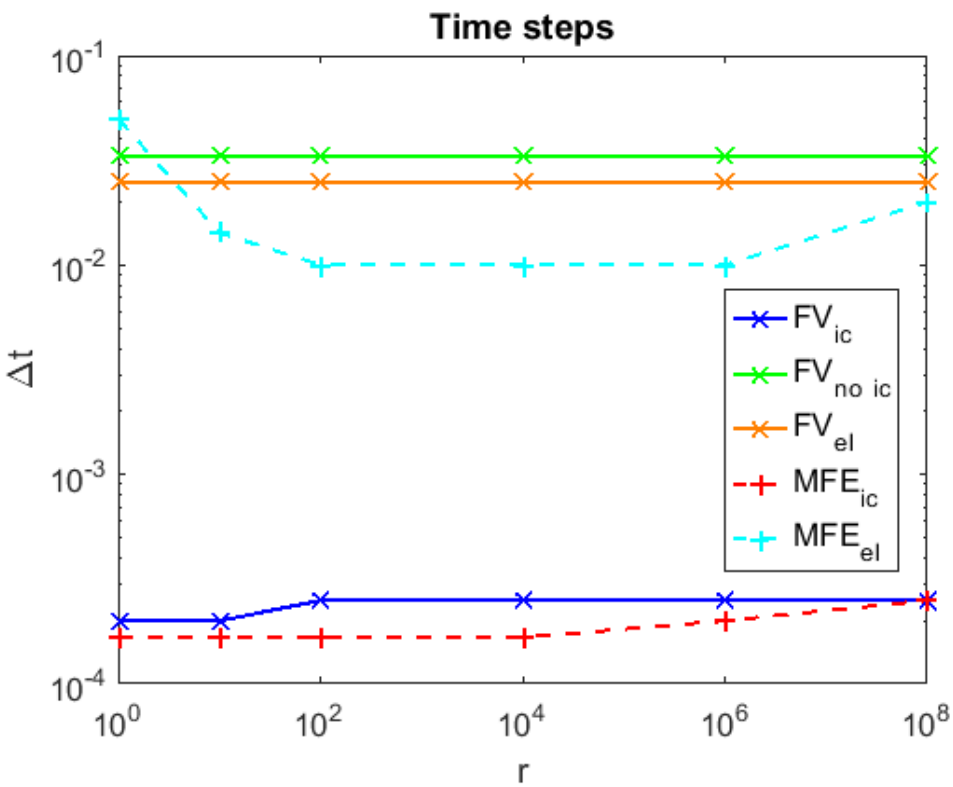


Figure 26: Time steps used for the simulation series of test case 4. The allowed step sizes are

almost constant for each of the combinations.

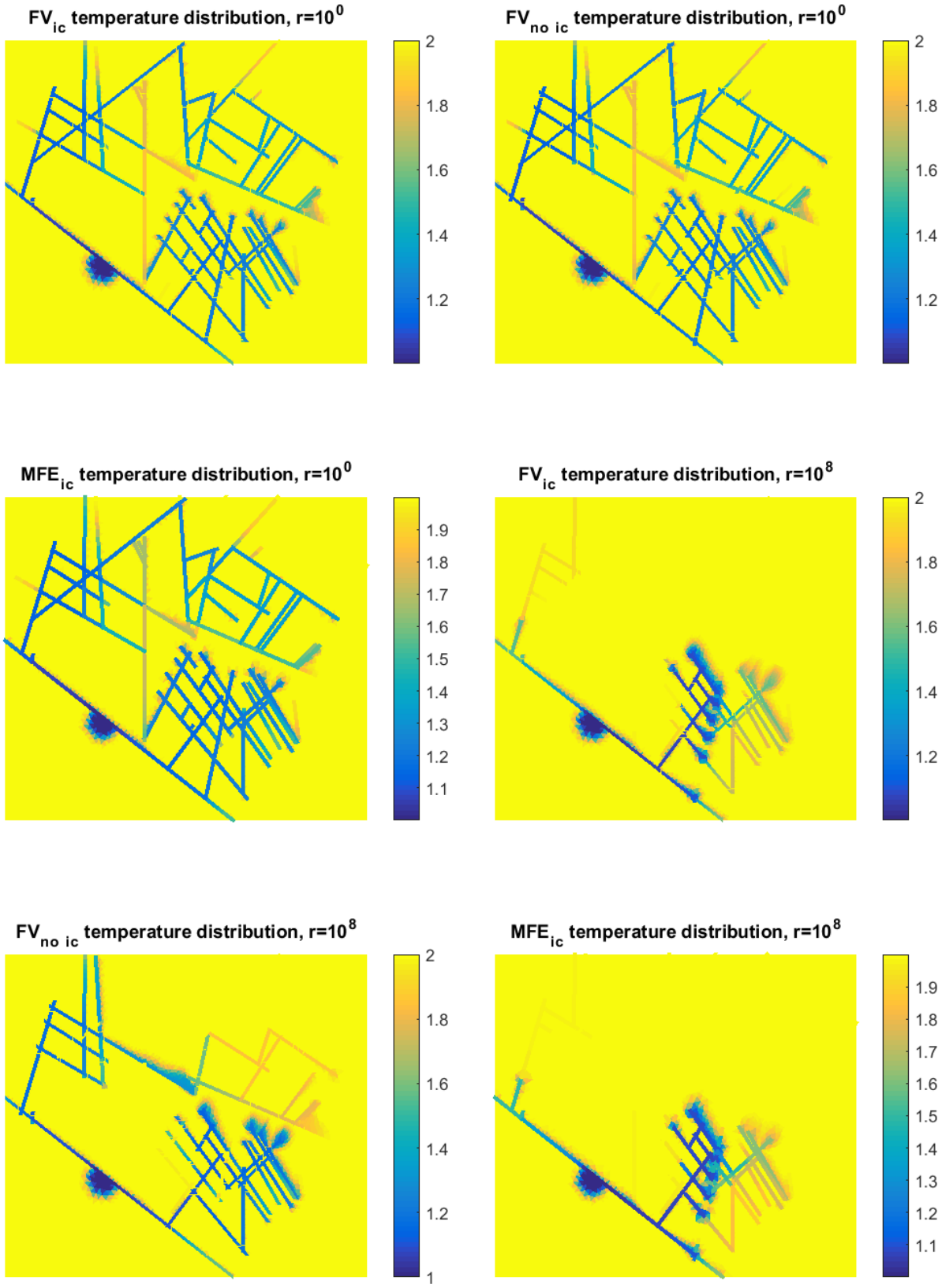
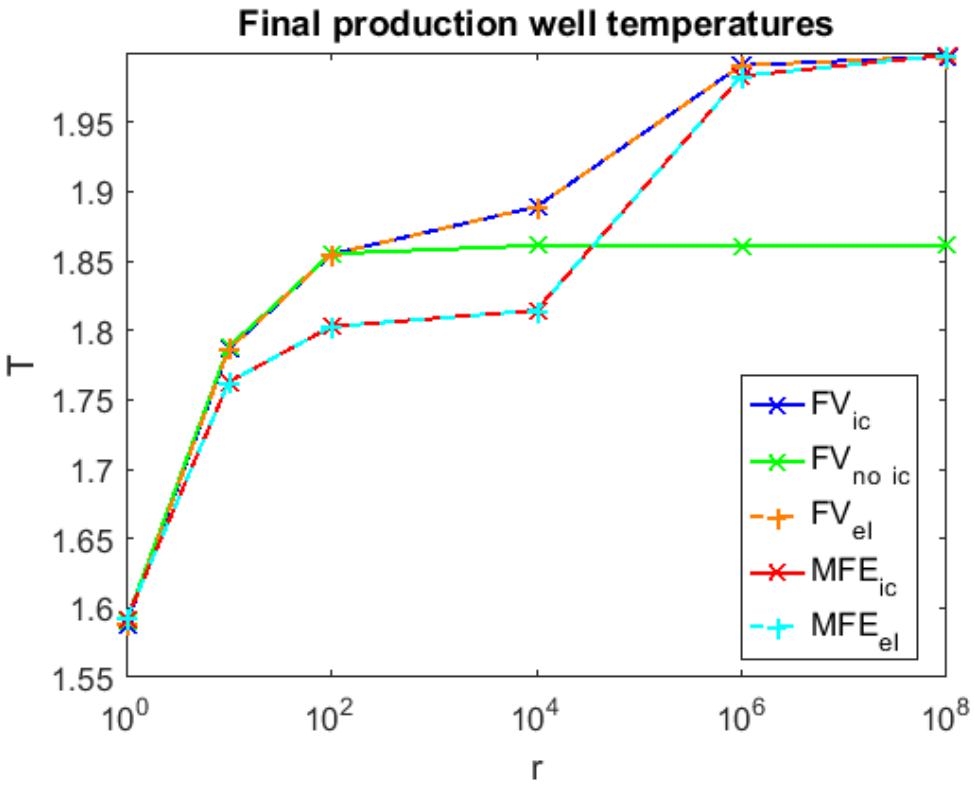


Figure 27: Temperature distributions of case f for the two most extreme fracture permeability

ratios.


Final production well temperatures
T
r
1.55
1.6
1.65
1.7
1.75
1.8
1.85
1.9
1.95
10^0
10^2
10^4
10^6
10^8
FV_{ic}
$FV_{no\ ic}$
FV_{el}
MFE_{ic}
MFE_{el}

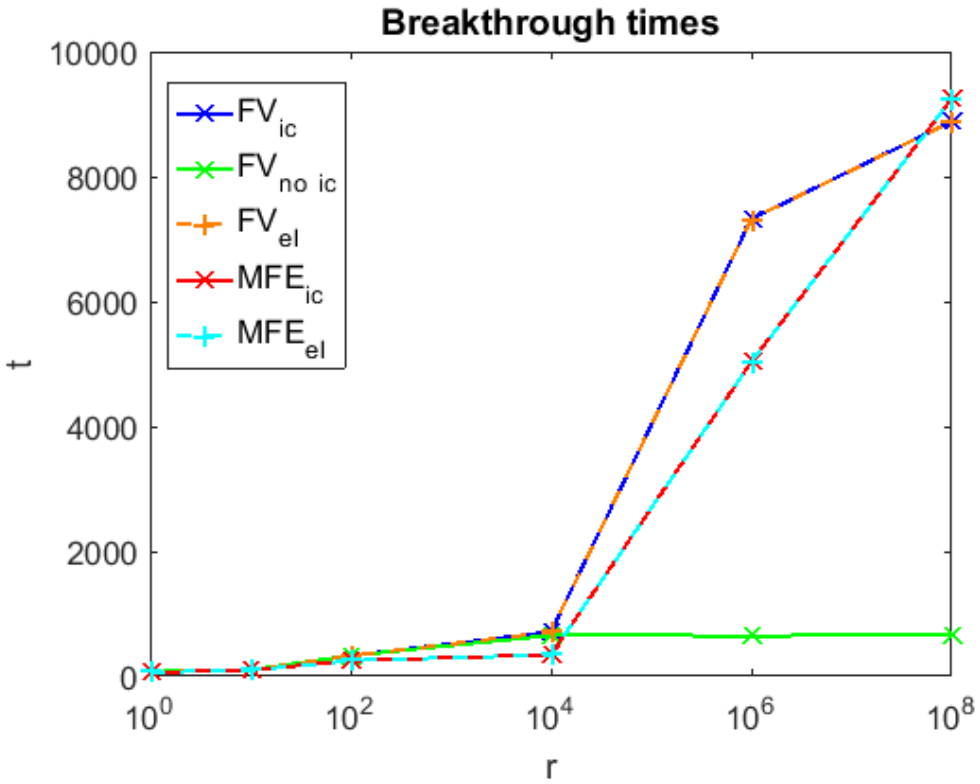

Breakthrough times
t
r
0
2000
4000
6000
8000
10000
10^0
10^2
10^4
10^6
10^8
FV_{ic}
$FV_{no\ ic}$
FV_{el}
MFE_{ic}
MFE_{el}

Figure 28: Final temperatures and breakthrough times of all combinations and fracture permeability ratios for the simulation of case f.

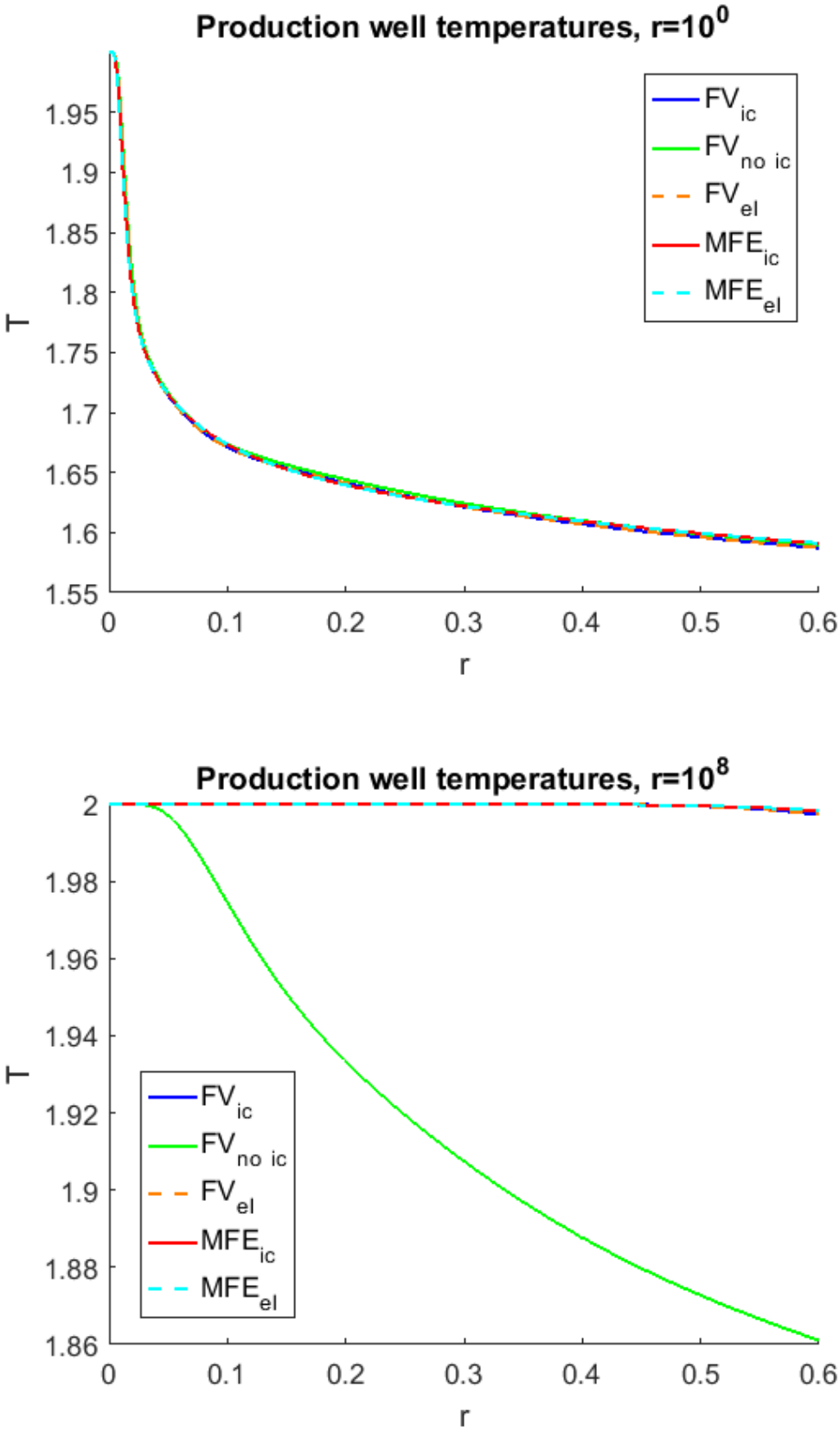
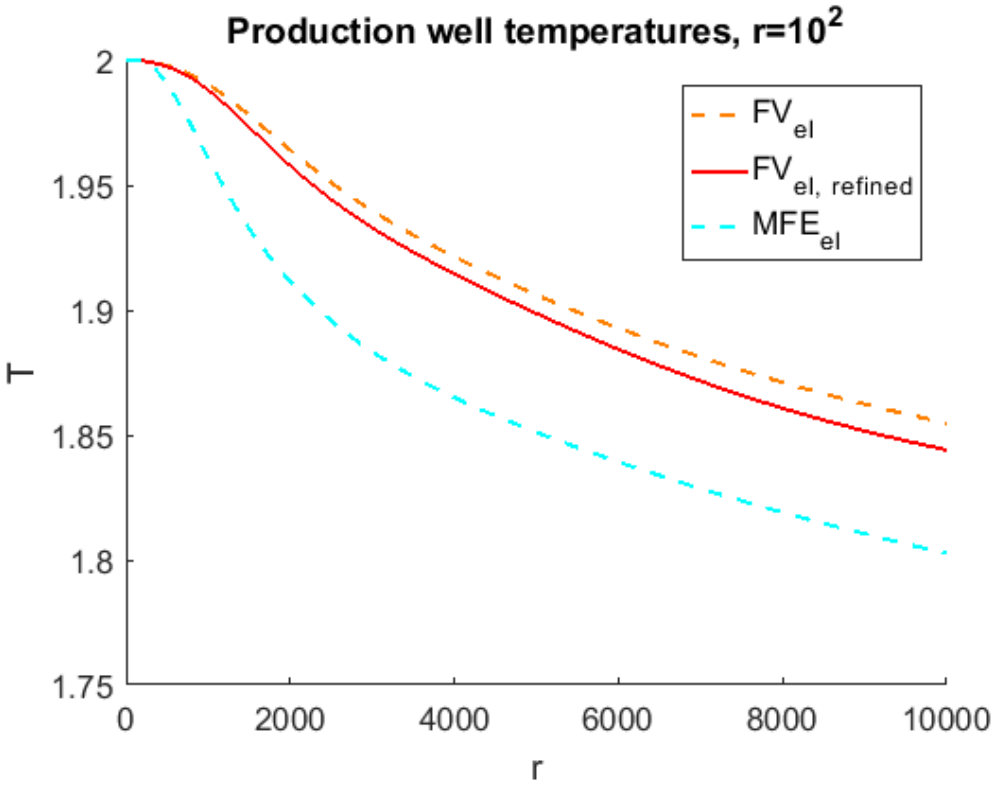


Figure 29: Production well temperature curves for the most extreme fracture ratios.



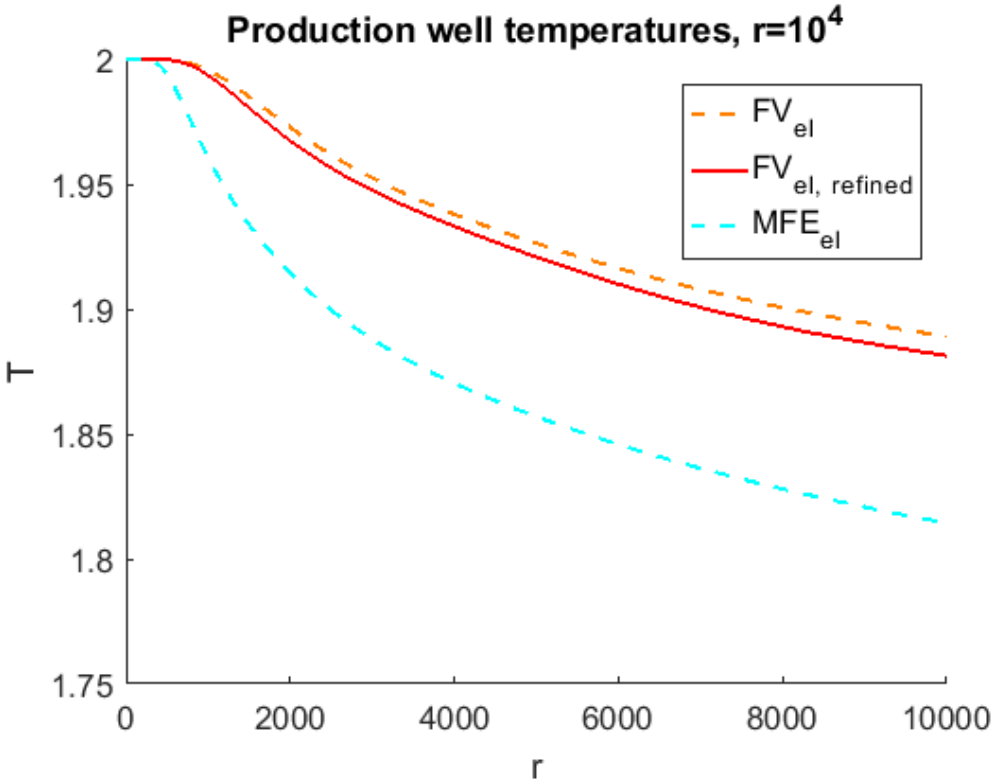


Figure 30: Production well temperatures for the eliminated combinations and a refined FV_{el}.

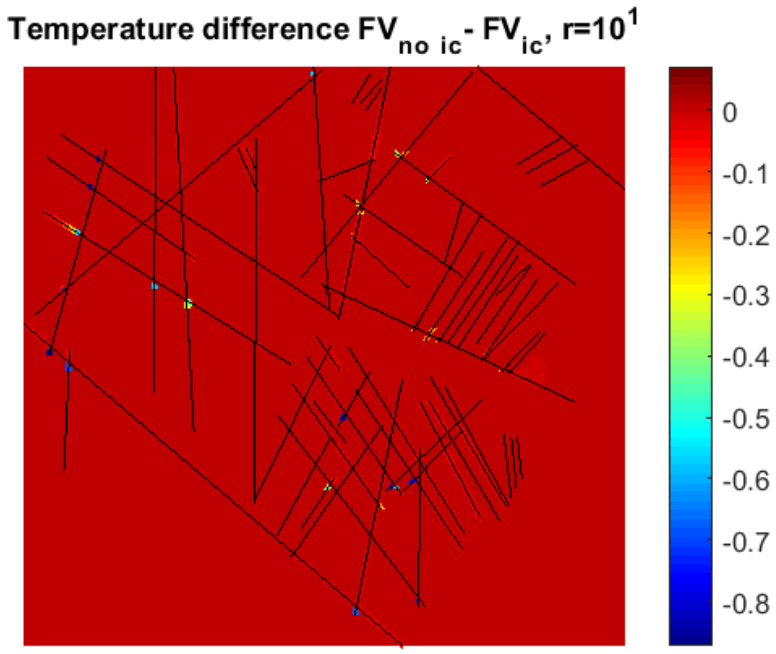


Figure 31: Difference between two of the FV combinations with fractures superimposed as thin black lines. Negative values mark regions to which the cold water has propagated in the $FV_{no\ ic}$ but not in the FV_{ic}.

The first comment about this test case is that it demonstrates the methods' applicability to complex geometries. As can be seen in the comparison study, not all methods for fractured porous media handle this geometry. What is more, the equidimensional model does not cope either, due to the vast number of cells and degrees of freedom needed to for a two dimensional grid of this complexity. This shows that there is a need for simplified models (e.g. co-dimension one approach) if realistic flow scenarios are to be solved numerically, and that the two methods presented in this thesis are of the ones capable of the simulations.

This test case is so complex, the explanations for the differences observed are not immediately obvious. Nevertheless, based on the lessons learned from the more synthetic cases discussed above, we feel quite confident in attributing some of the observations to the same reasons as in the previous cases. This means that no new effects will be identified from this example, but that we generalize the insights from the previous sections to the realistic scenarios we are really interested in modelling.

The two full methods agree well for both low and high r values. However, the final production well temperature and breakthrough time discrepancy in the intermediate r range is considerable.

The production well temperature plots with both coarse and refined solutions in Figure 30 confirm that there is a difference between the methods and suggest that the coarse FV, at least, is not quite converged, but not very far off. We note that the earlier arrival of injection fluid resembles what was observed from the temperature monitoring in case c.

While we might be tempted to suggest further investigations on this point, we also point to the plots of the final temperature distribution in Figure 27. They show that although the tracer seems to be transported more rapidly through the fracture network in the MFE simulations (the tracer concentration is significantly higher around the production well for $r = 10^2$, for example), the general qualitative flow pattern seems to be very similar. However, we have not found much information in these plots to help pinpoint the cause of the discrepancy. Our main leads to explain it is the intersection cell effect discussed in relation to test case b, where we observed a leakage out of the fractures right upstream of the intersections for the MFE simulations and the too early arrival from case c. For now, we leave the matter unresolved noting that nothing remarkable was found for this geometry in the benchmark study, demonstrating again how much more sensitive transport studies are compared to pure pressure comparisons.

We again have perfect fit between the full methods and the corresponding elimination combinations. The allowed time steps of Figure 26 show a gain of two orders of magnitude. We conclude from this that the elimination approach seems to work as hoped for any test case geometry, without loss of accuracy and with huge run time improvement potential.

The approach of eliminating intersection cells between flow and transport simulation was inspired by the observation of the breakdown of the $\mathrm{FV}_{\mathrm{no\ ic}}$ in certain cases. By applying it to the MFE method as well, we demonstrate that it is not restricted to the particular FV method and implementation. Thus, it can be used to relax the time step restrictions for transport simulations based on flux fields produced by any DFM method whit intersection cells. The close fits between the full combinations and the eliminated ones observed throughout this thesis, suggest that it can be performed with virtually no loss of accuracy of the resulting temperature field. There is of course the matter of whether this holds for all classes of fractured porous media scenarios, which we cannot affirm at the present.

All relevant results show breakdown of the $\mathrm{FV}_{\mathrm{no\ ic}}$ for high r values. The Figure 27 $r = 10^6$ plots show that this is because the method does not capture the loss of fracture network connectivity for increased fracture permeability ratios. With completely different tracer patterns and production well data, we cannot but discourage its use in cases where crossing permeable and impermeable fractures influence the solution. At the lower end of the r range, however, both breakthrough times and final temperatures are in much better agreement with the other combinations.

To examine the behaviour a bit closer at these low permeability ratios, we have included the direct difference plot in Figure 31. Unlike for the breakthrough and final temperatures, we can now discern a slight difference between the $\mathrm{FV}_{\mathrm{no\ ic}}$ and the rest even at $r = 10$, which we attribute to the effect of the artificial $\mathrm{FV}_{\mathrm{no\ ic}}$ fluxes seen in test case b. We see temperature decrease in some of the dead ends of the fracture network where the $\mathrm{FV}_{\mathrm{ic}}$ tells us none should be. However, also taking into account the $r = 1$ plot of Figure 29, this does not alter the overall conclusion that the dissimilarities between $\mathrm{FV}_{\mathrm{no\ ic}}$ and the full method do not lead to major errors for low values of r. Unless one of the artificial fluxes should happen to be of critical importance, e.g. by substantially altering the local behaviour around a well, the $\mathrm{FV}_{\mathrm{no\ ic}}$ combination without intersection cells seems to be close enough to the full combination for most purposes also in complex cases of (near) uniform fracture permeability.

5. Conclusion

We have investigated a Finite Volume and a Mixed Finite Element discrete fracture-matrix method for flow in fractured porous media. In addition to comparing the pressure fields, we have performed transport simulations and compared the resulting temperature distributions. We have given particular attention to the fracture intersections, and examined the quality of our proposed procedure for eliminating the intersection cells between the flow and the transport simulation. The test cases presented and discussed lead to four main notions about the elimination procedure and the flow methods.

The elimination procedure produces no discernible additional error compared to the full method. It offers an improvement of two to three orders of magnitude in time step length and condition number for explicit and implicit temporal discretizations, respectively. The procedure can in principle be applied to other flow methods as well as those presented above.

The two methods presented are well suited to simulate flow in fractured porous media. They handle every geometry feature we have tried out and high permeability ratios for both permeable and blocking fractures of very small apertures. No problems are detected as we move to more complex geometries.

The transport simulations reveal substantial deviations for the MFE method in cases where the fractures have different permeabilities higher than that of the matrix. We have not concluded as to the cause of this behaviour.

In cases where the intersections between blocking and permeable fractures are blocking, we strongly discourage the practice of eliminating intersection cells before the flow simulation. Especially if one proceeds to transport simulations, this may yield severe errors.

6. Sources

Bear, J., 2010. *Modeling Groundwater Flow and Contaminant Transport.* Dordrecht: Springer.

Bear, J. & Bachmat, Y., 1991. *Introduction to Modelling of Transport Phenomena in Porous Media.* Dordrecht: Kluwer Academic Publishers.

Bernardi, C., Maday, Y. & Rapetti, F., 2005. *Basics and some applications of the mortar element method.* s.l.:GAMM-Mitteilungen.

Blazek, J., 2005. *Computational fluid dynamics: principles and applications.* Amsterdam: Elsevier Science.

Boon, W. M. & Nordbotten, J. M., n.d. Robust Discretization of Flow in Fractured Porous Media. *submitted, available online: arXiv:1601.06977*.

Class, H., 2007. *Models for non-isothermal compositional gas-liquid flow and transport in porous media.* [Online]
Available at: http://dx.doi.org/10.18419/opus-299

Flemisch, B. et al., 2016. Benchmarks of single-phase flow in fractured porous media, draft manuscript.

Flemisch, B. et al., 2011. Dumux: DUNE for Multi-phase, component, scale, physics: Flow and Transport in Porous Media. *Advances in Water Resources*.

Karimi-Fard, M., Durlafsky, L. J. & Aziz, K., 2004. An efficient discrete-fracture model applicable for general-purpose reservoir simulators. *Society of Petroleum Engineers Journal*, pp. 227-236.

Lie, K.-A., 2015. An Introduction to Reservoir Simulation Using MATLAB: User guide for the Matlab Reservoir Simulation Toolbox (MRST). *SINTEF ICT*.

Liseikin, V. D., 1999. *Grid Generation Methods.* Dordrecht: Springer.

Matthäi, S. K. & Belayneh, M., 2004. Fluid flow partitioning between fractures and a permeable rock matrix. *Geophysical Research Letters*.

Nield, D. A. & Bejan, A., 2006. *Convection in Porous Media.* s.l.:Springer.

Nordbotten, J. M. & Celia, M. A., 2012. *Geological Storage of CO2 Modelling Approaches for Large-Scale Simulation.* Hoboken: Wiley.

Sandve, T. H., Berre, I. & Nordbotten, J. M., 2012. An efficient multi-point flux approximation method for Discrete Fracture-Matix simulations. *Journal of Computational Physics*, p. 3784–3800.

Thiele, M. R., 2001. *Streamline Simulation.* Schloss Fuschl, 6th International Forum on Reservoir Simulation.

Zimmermann, R. W. & Böðvarsson, G. S., 1994. *Hydraulic Conductivity of Rock Fractures*, s.l.: Lawrence Berkeley National Laboratory.

7. Attachment: Benchmarks of single-phase flow in fractured porous media (Preprint)

The author's contribution to the attached benchmark paper was to provide the code for error evaluation, make the grids and perform the simulations for the two methods of this thesis for all cases except the last case for the MFE method, to write the Chapter 3.2 method description of the FV method and to provide Figure 2. We are much obliged to Professor Flemisch and the rest of the authors for agreeing to letting us attach a preprint of the paper here.

Noname manuscript No.
(will be inserted by the editor)

Benchmarks of single-phase flow in fractured porous media

Bernd Flemisch, Inga Berre, Wietse Boon, Alessio Fumagalli, Nicolas Schwenck, Anna Scotti, Ivar Stefansson and Alexandru Tatomir

the date of receipt and acceptance should be inserted later

Abstract Not yet written.

1 Introduction

Many fields of applications for porous media flow include geometrically anisotropic inclusions and strongly discontinuous material coefficients which differ in orders of magnitude. If the extension of those heterogeneities is small in normal direction compared to the tangential directions, e.g., long and thin, those features are called fractures. Examples which include such fractured porous-media systems in earth sciences include reservoir engineering, groundwater-resource management, carbon capture and storage (CCS), oil and gas recovery, radioactive-waste reposition, coal bed methane migration in mines, geothermal engineering and hydraulic fracturing.

The analysis and prediction of flow in fractured porous-media systems is important for all the aforementioned applications. Experiments are usually too expensive and time consuming to satisfy the demand for fast but accurate decision making information. In this sense, many different conceptual and numerical models of flow in fractured porous-media systems can be found in the literature. Even though fractured porous-media systems have been of interest to modelers for a long time, they still represent challenges for simulators. However, even in the time of large supercomputers with massive parallel

Bernd Flemisch, Nicolas Schwenck
Department of Hydromechanics and Modelling of Hydrosystems, Universität Stuttgart, Pfaffenwaldring 61, 70569 Stuttgart, Germany, E-mail: bernd@iws.uni-stuttgart.de

Alexandru Tatomir
Department of Applied Geology, Geosciences Center, University of Göttingen, Goldschmidtstrasse 3, 37077 Göttingen, Germany, E-mail: alexandru.tatomir@geo.uni-goettingen.de

Alessio Fumagalli, Inga Berre, Wietse Boon and Ivar Stefansson
Department of Mathematics, University of Bergen, Allégaten 41, 5007 Bergen, Norway, E-mail: alessio.fumagalli@uib.no

computing power, the computational efficiency, and implicitly the economic efficiency, plays an important role in the evaluation of simulation software.

Another key aspect for the translation from model concepts (fine and coarse scale) towards a scale for the size of actual problems (field scale) is that in most cases the exact structure on a scale larger than the Darcy scale is unknown. It is usually approximated by taking small samples of the subsurface structure at specific probing sites. This can be done by looking at cores or bore-hole images or by doing seismic surveys over a larger scale and then estimating the field scale structure. Because of the resulting uncertainty, stochastic modeling [1], has to be taken into account. This leads to the requirement of even faster and more efficient flow models.

During the last 70 years, different modeling approaches have been developed and gradually improved by many working groups around the globe having different interests. Comprehensive reviews can be found in [4, 8, 21, 30, 34, 38]. Roughly, the fractured porous media systems are classified in two broad categories: discrete fracture matrix models (DFMs) and continuum fracture models. We will only look at DFMs within this article.

The DFMs consider flow occurring both in the fracture network and the surrounding rock matrix. They account explicitly for the effects of individual fractures on the fluid flow. An efficient way to represent the fractures in DFMs is the hybrid-dimensional approach, e.g. [20, 10, 33], where fractures are discretized with elements of co-dimension one with respect to the surrounding matrix, for example linear, 1D elements in two-dimensional settings.

Among the different fracture model concepts, DFMs are considered the most accurate as they use explicit representations of the fractures. The accuracy of DFMs is usually paid with higher computational effort invested in both grid generation and flow simulation. A standard way to reduce computational time is the assumption of lower-dimensional fracture models. We assume that the conceptual co-dimension one model for the fractures is the most efficient and thus, only use lower-dimensional discretizations for the fractures. In addition, it is hard to obtain an exact knowledge of the geometrical (e.g., fracture length, width, orientation, density, connectivity, etc.) and hydraulic properties of the fractured system. Nevertheless, the recent advances in geophysical, hydraulic and tracer reservoir characterization techniques complemented by sub-seismic scale measurements (i.e., pore- and core-scale characterizations) and advanced statistical or geostatistical fracture generators allow an accurate and reliable reconstruction of the fracture network [1]. We assume the fracture properties to be known and input for our models.

The flow regimes within the fractures reach from classical Darcy flow to free flow, e.g., described by Navier-Stokes equations, or with Forchheimer or Brinkman equations as intermediate cases. Depending on the fracture-matrix permeability ratios, fracture geometry, fracture density and fracture connectivity, different flow regimes can be identified [25]. Here, fracture density is a measure for the number of fractures in the control region and fracture connectivity a measure of how well the fractures are interconnected. The flow regime can be dominated by flow through the fractures, flow trough the ma-

trix or both. We look only at Darcy flow type problems with different fracture importance, i.e., fracture-matrix permeability ratios.

Motivation The main focus of this work is to investigate the strength and weaknesses of [37,36], a novel hybrid-dimensional fracture XFEM implementation which is based on [24], similar to [6]. This non-conforming discrete fracture modeling approach is compared against a conventional, continuous pressure, conforming hybrid-dimensional DFM [40] (Box-DFM) and a reference solution. Four specific two-dimensional benchmark problems with varying degrees of geometrical complexity are shown for the validation and evaluation of the model. The models are tested with respect to accuracy and computational efficiency to simulate flow in fractured porous systems. Besides improving the overall understanding of fracture modeling techniques, our research helps modelers selecting an appropriate discretization method for a given specific problem.

2 The model problem

We face the model problem of stationary, incompressible single-phase flow through a porous medium,

$$\mathbf{u} = -\mathbb{K}\,\mathbf{grad}\,p, \tag{1a}$$

$$\operatorname{div}\mathbf{u} = q, \tag{1b}$$

in an open bounded domain $\mathcal{D} \subset \mathbb{R}^N$, subject to boundary conditions

$$p = p_{\mathrm{D}} \text{ on } \partial\mathcal{D}_{\mathrm{D}}, \tag{1c}$$

$$\mathbf{u}\cdot\mathbf{n} = q_{\mathrm{N}} \text{ on } \partial\mathcal{D}_{\mathrm{N}}, \tag{1d}$$

with $\partial\mathcal{D} = \overline{\partial\mathcal{D}_{\mathrm{D}} \cup \partial\mathcal{D}_{\mathrm{N}}}$ and $\partial\mathcal{D}_{\mathrm{D}} \cap \partial\mathcal{D}_{\mathrm{N}} = \emptyset$. Although $\mathbb{K}$ and p stand for absolute permeability and pressure, everything can be easily adapted to the more general case of a mobility and a potential, respectively. Moreover, an additional storage term, appearing in (1b) in the case of a compressible fluid phase, usually does not pose any difficulties.

Let us assume that $\mathcal{D}$ contains several fractures, that all together constitute a single domain Γ of spatial dimension N such that $\Gamma \subset \mathcal{D}$, which is a possibly unconnected, open subset of $\mathcal{D}$. The surrounding porous rock, namely, the remaining part of $\mathcal{D}$, is called $\Omega = \mathcal{D} \setminus \overline{\Gamma}$. Assuming that the aperture d at each point of Γ is small compared to other characteristic dimensions of the fractures, the full-dimensional domain Γ can be reduced to the $(N-1)$-dimensional fracture network γ. For a formal presentation of this dimensional reduction, see [24]...

3 Participating discretization methods

Within this section, the discretization methods are described that participate in this benchmark study. The purpose of this article is the comparison of well-known, established and/or at least published methods. Therefore, only the most significant aspects of each method are summarized. We do not show a comparison against analytical solutions here. The analysis of the methods and, for example, a proof for optimal convergence can be found in the references listed below. A summary of all participating methods is provided in Table 1. In the sequel we make use of *d.o.f.* indicating the degrees of freedom associated to a specific method.

acronym	**d.o.f.**	**frac-dim**	**conforming**	**p-cont.**
Box-DFM	p (vertices)	dim-1	yes	yes
CC-DFM	p (elements)	dim-1	yes	no
EDFM	p (vertices)	dim-1	no	yes
Mortar-DFM	p (elements)	dim-1	geometrically	no
P-XFEM	p (vertices)	dim-1	no	no
D-XFEM	p (elements), $\boldsymbol{u}$ (faces)	dim-1	no	no
MFD	p (elements), $\boldsymbol{u}$ (faces)	dim	–	no

Table 1: Participating discretization methods.

3.1 Vertex-centered, continuous-pressure, conforming lower-dimensional DFM (Box-DFM)

The lower-dimensional representation of fractures allows easier mesh generation in comparison to the equi-dimensional approach, as it circumvents the appearance of very small elements when discretizing the interior of the fracture (i.e., within the fracture width). Implicitly this leads to shorter computing times. Furthermore, the conforming mesh generation algorithm accounts for the geometrical characteristics of the fracture system. Conform meshing implies that the fractures are discretized with a set of linear elements (in a 2D domain) and that they are also the edges of the triangular finite elements.

The spatial discretization in Box-DFM is performed with the box method (vertex-centered finite volume method) [20] which combines the advantages of finite element and finite volume grids, allowing unstructured grids and being locally conservative. Figure 1 illustrates a two-dimensional representation of the dual-grid with two finite elements E_1 and E_5 are sharing the same edge (ij_1) representing a lower-dimensional fracture with the aperture ϵ_{ij}. The main characteristics in terms of the fractured system is that the pressure is required to be continuous, in particular also in those vertices which control volumes share fracture and matrix regions.

The Box-DFM method is implemented in the free-open-source numerical simulator DuMu^{x}. A detailed description of the conceptual, mathematical, numerical and code implementation is published in [40]. The Box-DFM simulation code used for the benchmark studies is publicly available under `https://git.iws.uni-stuttgart.de/dumux-pub/Flemisch2016a.git`.

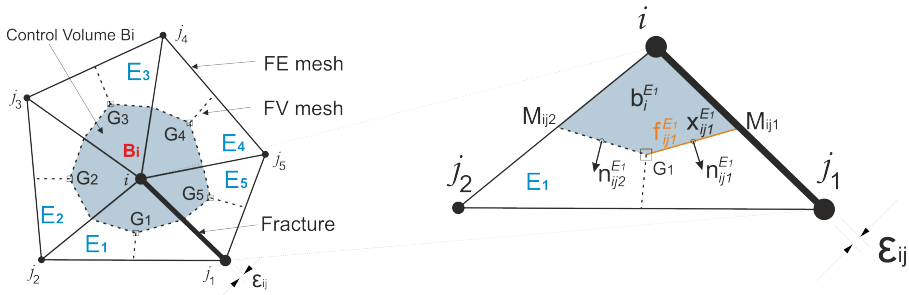


Fig. 1: Conceptual representation of the Box-DFM method: (left-hand side) The dual finite element and finite volume mesh from which the control volume B_i around node i is created. Node i is surrounded by nodes $\{j_1, j_2, j_3, j_4, j_5\}$, where segment ij_1 represents both a fracture and a shared FE edge; (right-hand side) Sub-control volume (SCV) $b_i^{E_1}$ in element E_1 has barycenter G_1 and the mid-points of the edges ij_1 and ij_2 are M_{ij1}, respectively M_{ij2}. The SCV face f_{ij1}^{E1} is the segment $\overline{G_1 M_{ij1}}$ which contains the integration point x_{ij1}^{E1} where the normal vector $\mathbf{n}_{ij1}^{E1}$ is applied.

3.2 Cell-centered, discontinuous-pressure, conforming DFM (CC-DFM)

The control volume finite difference method uses a two-point flux approximation based on the cell-centre pressure values for the evaluation of the face fluxes. The domain is partitioned with fractures coinciding with the interior faces between matrix cells just as described in Section 3.1. We approximate the flux over the face between matrix cells i and j as

$$\mathbf{u}_{ij} = T_{ij}(p_i - p_j), \tag{2}$$

where p_i and p_j are the pressures in the neighbouring cells and T_{ij} is the face transmissibility, computed as the harmonic average of the two half transmissibilities corresponding to the face and the two cells. The half transmissibility of cell-face pair i is in turn given as

$$\alpha_i = \frac{A_i \mathbf{n}_i \cdot K_i}{\mathbf{d}_i \cdot \mathbf{d}_i} \mathbf{d}_i, \tag{3}$$

where A_i and $\mathbf{n}_i$ are the area and unit normal vector of the face, K_i is the permeability assigned to the cell and d_i is the distance vector from cell centre

to face centroid. In addition to the unknowns given at the centroids of the matrix cells, unknowns are associated to the centroids of the fracture cells. The fracture cells are assigned apertures, which multiplied with the length give the volume of these cells. The aperture is also used to produce hybrid faces for the matrix-fracture interfaces. These faces, parallel to the fracture but displaced half an aperture to either side, enable us to compute the half transmissibilities corresponding to these faces. These faces are indicated by the dashed blue lines in Figure 2, where the computational domain is superimposed on the geometrical grid. The result is a hybrid grid with fractures which are lower dimensional in the grid, but equidimensional in the computational domain at the cost of a small matrix volume error corresponding to the overlap of the matrix cells with the fracture cells.

The intermediate fracture intersection cell drawn with dashed red lines in Figure 2 is excluded, leading to direct coupling of the fracture cells neighbour to the intersection. The purpose of this is both to obtain a smaller condition number and to avoid severe time-step restrictions associated with small cells in transport simulations. To each new face between cell i and j, we assign face transmissibilities calculated using the star delta transformation as described in [22]:

$$T_{ij} = \frac{\alpha_i \alpha_j}{\sum_{k=1}^{n} \alpha_k}, \tag{4}$$

with n denoting the number of fracture cells meeting at the intersection. For an extension of the method to non-isotrop media using a multi-point flux approximation, we refer to [35].

3.3 Continuous-pressure, non-conforming embedded DFM (EDFM)

Recently, non-conforming methods for the treatment of lower-dimensional fractures were developed, for example in [27,26,18], to avoid the construction of complex and time consuming grids which represent explicitly the fractures. They are mostly used in the context of single and multi-phase flow simulations for petroleum engineering applications and require the normal fracture permeability to be orders of magnitude higher than the matrix permeability, like in enhance reservoir exploitation and fractures stimulation. In this field of applications corner-point grids are normally employed to describe the layers, *e.g.* different rock type, of the reservoir. An adaptation of the computational grid to the fractures could be unaffordable for real cases. The numerical method belongs to the family of two-point scheme, where a one-to-one connection, through the transmissibility concept, between the degrees of freedom is considered [9]. References of embedded discrete fracture method (EDFM) can be found, for example, in [23,32,28,31,7,15].

In practice, the mesh of the fractures are generated on top of the rock grid so that, for each rock cell that they intersect, contains one fracture cell per fracture. Intersections among fractures are computed, without affecting

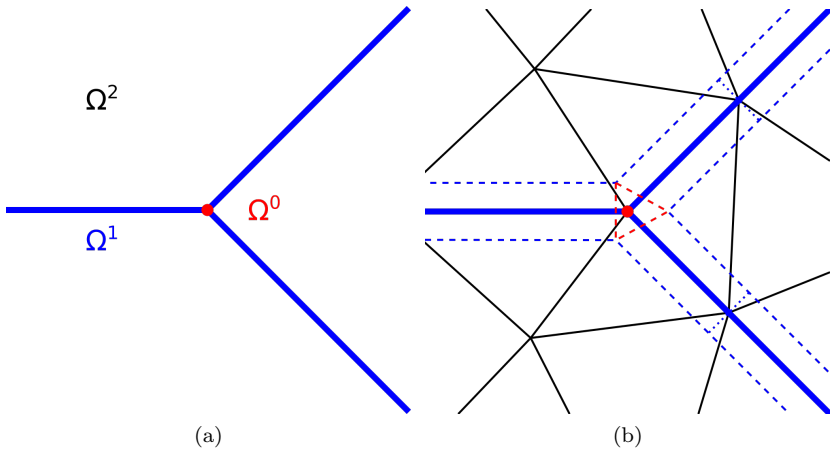


Fig. 2: (a) Conceptual decomposition of the domain according to element dimension with the matrix depicted in black, fractures in blue and their intersections in red. (b) The computational domain of the CC-DFM. Dashed lines are faces of the fracture cells.

the creation of the grids of fractures and rock, which are used to compute approximate transmissibilities between different fracture cells. See Figure 3 as an example. A degree of freedom, that represents a pressure or a saturation

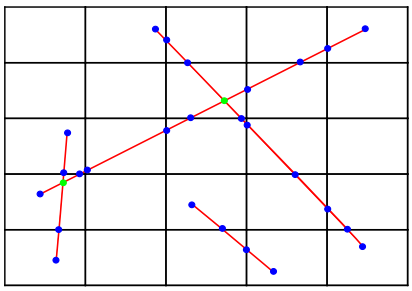


Fig. 3: Example of meshes, for both fractures and rock matrix, suited for EDFM. The rock matrix is considered as a background mesh. Each fracture cells is represented by two blue dots and the green dots are the non-matching intersection among fractures.

value, is assigned to each matrix cell and to each fracture cell. This means that transmissibilities between matrix and fracture cells, as well as those between different fracture cells, need to be computed. The transmissibility between a fracture cell and a matrix cell T_{fm} and the half-transmissibility T_i, related to the fracture i, between two intersecting fracture cells are computed, respec-

tively, through the following approximate expressions

$$T_{fm} = A\frac{\mathbf{n}_f^\top \mathbb{K} \cdot \mathbf{n}_f}{d_{f,m}} \quad \text{and} \quad T_i = s\frac{k_i d_i}{d_{i,s}}.$$

Where A is the measure of the fracture cell in the current rock cell, $\mathbf{n}_f$ is the normal of the fracture cell and $d_{f,m}$ is an average distance between the fracture cell and the matrix cell [23]. For the fracture-fracture transmissibility s is the measure of the intersecting segment, k_i the scalar permeability of the fracture, d_i the aperture and $d_{i,s}$ is the average distance between the fracture cell and the intersecting segment. The standard harmonic average is considered to compute the transmissibility between the two fracture cells. Standard formulae for fracture-fracture as well as matrix-matrix transmissibility are computed in a two-point flux approximation framework. It is worth to notice that in [41] an extension of EDFM called Projection-based EDFM (pEDFM) is able to handle also low permeable fractures.

3.4 Cell-centered, discontinuous-pressure, geometrically-conforming mortar DFM (Mortar-DFM)

Not yet written.

3.5 Discontinuous-pressure, non-conforming primal XFEM (P-XFEM)

The primal XFEM method participating in this benchmarky study is described in detail in [36], see also [12,37]. The method is based on the hybrid-dimensional problem formulation investigated in [24], where conditions for the coupling between fracture and matrix are derived:

$$\{\!\{\mathbf{u}_\mathrm{m} \cdot \mathbf{n}\}\!\}_\gamma = k_{\mathrm{f,n}} / d \, [\![p_\mathrm{m}]\!]_\gamma \tag{5a}$$

$$\xi_0 \, [\![\mathbf{u}_\mathrm{m} \cdot \mathbf{n}]\!]_\gamma = k_{\mathrm{f,n}} / d \left(\{\!\{p_\mathrm{m}\}\!\}_\gamma - p_\mathrm{f} \right) \tag{5b}$$

Here, the subscripts "m" and "f" indicate matrix and fracture quantities, while $\{\!\{\cdot\}\!\}_\gamma$ and $[\![\cdot]\!]_\gamma$ denote the average and the jump of a matrix quantity over the fracture γ, respectively.

The coupling conditions (5) can be used to define a source term for the fracture flow problem, while they yield an interface problem for the matrix domain. For the discretization of this interface problem, the methodology presented in [19] is used, which amounts to applying the extended finite element method. Together with an independent standard discretization of the lower-dimensional fracture problem, this yields a hybrid-dimensional, non-conforming primal XFEM-based method. Standard Langrangian $\mathbb{P}_1/\mathbb{Q}_1$ finite-element spaces are employed, i.e., the degrees of freedom are located at the vertices of the full-dimensional grid of the matrix Ω and the lower-dimensional grid of the fracture γ. A representative example of matrix and fracture grids is

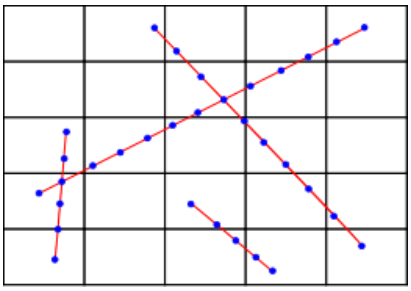


Fig. 4: Example of meshes, for both fractures and rock matrix, suited for P-XFEM. The fracture grid vertices are indicated by the blue dots.

illustrated in Figure 4. In comparison with Figure 3 and the EDFM method, the fracture grid vertices can be placed arbitrarily without taking into account the matrix grid. On the other hand, the method requires matching fracture branch grids in the form of vertices placed at the fracture intersections. In particular, special care has to be taken of intersecting and immersed fractures [37].

The method is implemented on top of the DUNE framework [2] and the discretization module DUNE-PDELab [3]. For the enrichment of the finite-element spaces in the context of XFEM, the modules DUNE-Multidomain and DUNE-Multidomaingrid are employed [29]. The simulation code for the XFEM approach and for the benchmarks studied here is publicly available under `https://git.iws.uni-stuttgart.de/dumux-pub/Flemisch2016a.git`.

3.6 Discontinuous-pressure, non-conforming dual XFEM (D-XFEM)

The dual XFEM method participating in his benchmark is based on [6]. The method, originally derived for a domain cut by one fracture, was further developed in [14], [16] to account for intersecting fractures with different permeabilities. The same equations and coupling conditions as for the primal XFEM are used, but in a dual formulation where Darcy law and mass conservation give rise to a saddle-point problem for the fluid mean velocity and pressure, both in the fracture and in the surrounding medium. The usual lowest order Raviart Thomas $\mathbb{RT}_0 - \mathbb{P}_0$ pair for velocity and pressure is enriched following [19] in the elements of the porous medium cut by a fracture, or in the elements of a fracture at the intersection with other fractures. Indeed, triangular/tetrahedral grids are arbitrarily cut by triangulated lines/surfaces in 2D and 3D respectively that can, in turn, intersect each other in a non-conforming way, as shown in figure 2.

The method has been implemented on the basis of the Getfem++ library, `http://download.gna.org/getfem/html/homepage/`, which provides support for the computation of the intersections and the quadrature on sub-elements thanks to an interface with QHull, .`http://www.qhull.org/`.

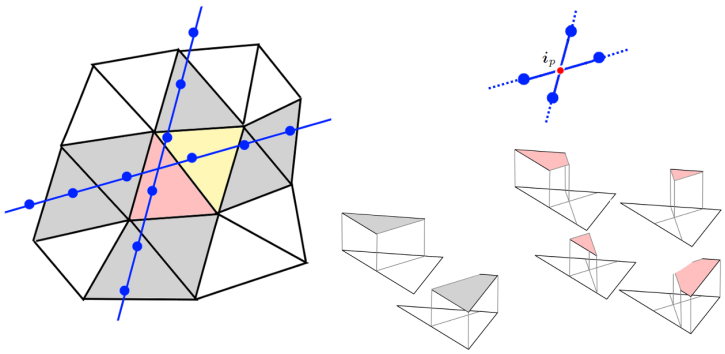


Fig. 5: A portion of the grid cut by two fractures: in the two dimensional case they can split the elements in two (grey), three (yellow), or four (red) independent parts, where the restrictions of the basis functions are defined. The fracture grids are irrespectiveof the bulk grid and of each other, i.e. the intersection point i_p is not a point of the grid.

3.7 Reference Solutions calculated with mimetic finite differences (MFD)

The reference solutions are computed on very fine grids that discretize both matrix and fractures by full-dimensional triangular or quadrilateral elements. A mimetic finite difference method [5,13] is used to discretize (1). The method is employed as it is implemented in DuMu^{x} 2.7 [11]. In particular, a mixed-hybrid approach is used to transfer the discrete saddle point problem in terms of cell pressures and face fluxes to a symmetric positive definite formulation with face-pressure degrees of freedom.

4 Benchmark Problems

This is the main section which compares the methods described above by means of four benchmark cases. First, in Section 4.1, we present a well established benchmark for groundwater flow [39] that contains two crossing, highly permeable fractures and a non-straight surface. The second benchmark case, Section 4.2, is based on [17] and shows a regular fracture network. After that, a small but complex fracture network exhibiting ending and intersecting fractures is investigated in Section 4.3. Finally, a case synthesized from a real application is considered in Section 4.4.

4.1 Benchmark 1: Hydrocoin

Within the international Hydrocoin project, [39], a benchmark for heterogeneous groundwater flow problems was presented. The domain setup is shown

in Figure 6 with the exact coordinates given in Table 2. We note that we have

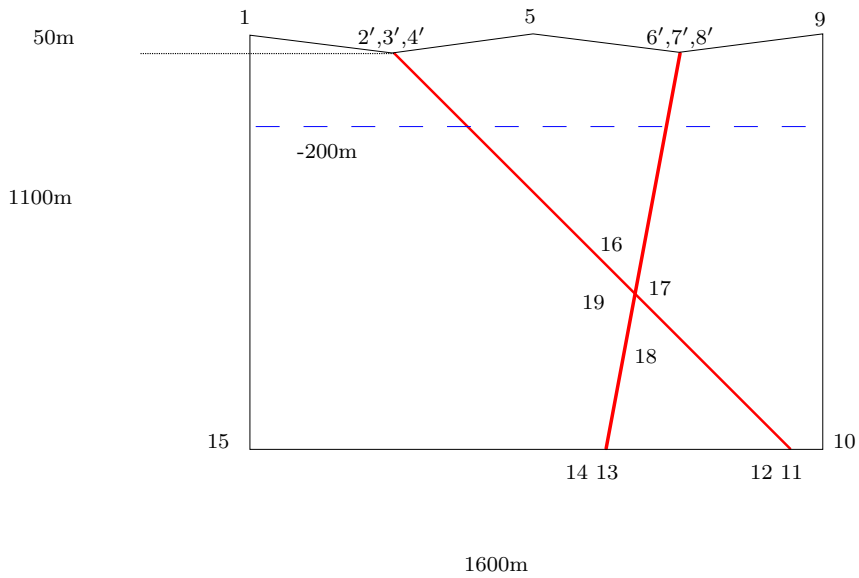


Fig. 6: Geometry of the modeled domain of the Hydrocoin test case 2, [39]. Modified node locations are indicated by numbers superscripted with ′. Boundary conditions are hydraulic head on top and Neumann no-flow on the other three sides of the domain.

Table 2: Coordinates of the numbered points in the modeled region of the problem depicted in Figure 6.

point	x (m)	z (m)	point	x (m)	z (m)
1	0	150	11	1505	-1000
2′	394.285714286	100.714285714	12	1495	-1000
3′	400	100	13	1007.5	-1000
4′	404.444444444	100.555555556	14	992.5	-1000
5	800	150	15	0	-1000
6′	1192.66666667	100.916666667	16	1071.34615385	-566.346153846
7′	1200	100	17	1084.03846154	-579.038461538
8′	1207.6744186	100.959302326	18	1082.5	-587.5
9	1600	150	19	1069.80769231	-574.807692308
10	1600	-1000			

slightly modified the original domain such that an equi-dimensional model can be run on exactly the same domain as the hybrid-dimensional models. This allows for an easier comparison of the solution values over the whole domain. In

particular, the plateaus close to the upper left and right corners 1 and 9 have been omitted. Moreover, the upper ends of the two fractures have been modified according to Figure7 which amounts to the changes of nodes 2–4 and 6–8. Finally, the position of nodes 16–19 has been recalculated with higher preci-

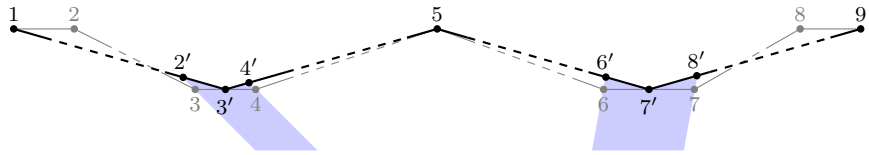


Fig. 7: Modifications of the Hydrocoin model domain compared to the original formulation [39]. The original upper boundary is drawn with gray thin lines, while thick black lines are used for the modified boundary. Modified node locations are indicated by numbers superscripted with $'$. The shaded regions show the upper parts of the two slightly extended equi-dimensional fractures.

sion. The hybrid-dimensional models don't take into account nodes 2,4,6,8 and 16–19 and combine nodes 11,12 and 13,14, since the two-dimensional fracture regions have been reduced to two intersecting straight lines.

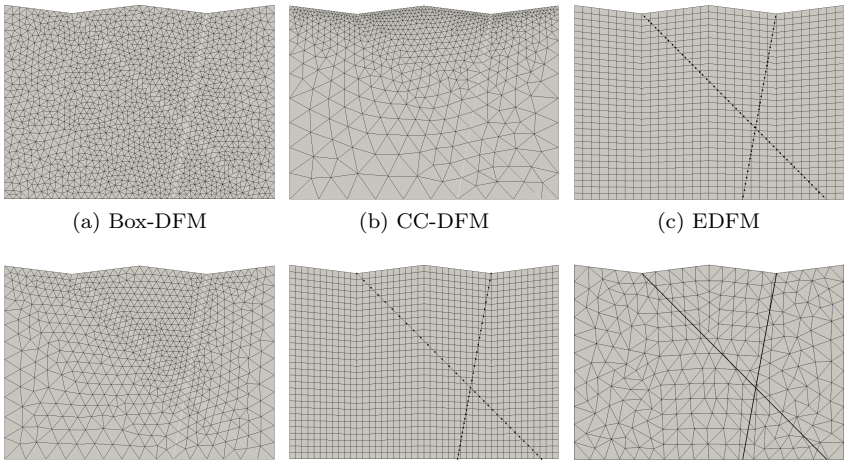
The boundary conditions are Dirichlet piezometric head on the top boundary and Neumann no flow on the other three boundaries. The hydraulic conductivity in the fracture zones is $10^{-6}\,\mathrm{m/s}$ and in the rock matrix $10^{-8}\,\mathrm{m/s}$, respectively. The inclination of the fracture zones has no influence on the permeability tensor and in lower-dimensional models, the normal and tangential permeabilities are exactly the same scalars.

Table 3 lists the number of degrees of freedom, matrix elements and fracture elements for all the participating methods. The corresponding grids are

method	**d.o.f.**	**matrix elems**	**fracture elems**
Box-DFM	1496	2863 triangles	74
CC-DFM	1459	1416 triangles	43
EDFM	1044	960 quads	84
Mortar-DFM	3647	1384	63
P-XFEM	1667	1320 quads	68
D-XFEM	3514	1132 triangles	160
MFD			

Table 3: Grids for Benchmark 1.

visualized in Figure 8.



(a) Box-DFM (b) CC-DFM (c) EDFM

(d) Mortar-DFM (e) P-XFEM (f) D-XFEM

Fig. 8: Benchmark 1: the grids used by the different methods.

The original benchmark shows the piezometric head distribution along five horizontal lines through the modeled domain. Here, we first show in Figure 9 the plot at a depth of 200 m, as indicated by the dashed line in Figure 6.

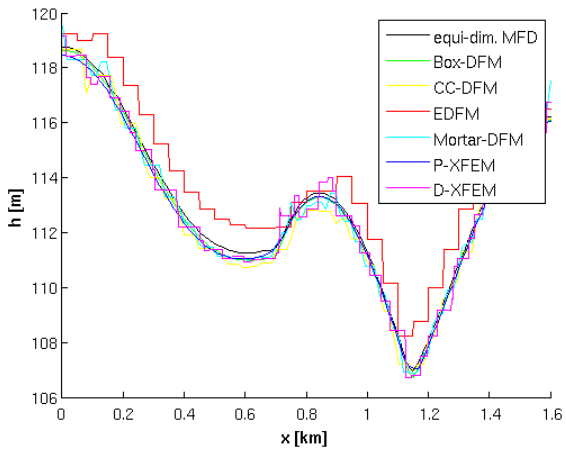


Fig. 9: Benchmark 1: pressure values along a horizontal line at a depth of 200 m.

Table 4 lists the discretization errors for the different methods, particularly, the error for the matrix domain and the one along the two fractures. Moreover,

method	**matrix error**	**fracture error**	**nnz/size^2**	**$\|\cdot\|_2$-condition**
Box-DFM	9.3e-3	3.3e-3	4.5e-3	5.4e3
CC-DFM	1.1e-2	1.1e-2	2.7e-3	3.5e4
EDFM	1.5e-2	8.3e-3	4.7e-3	3.9e4
Mortar-DFM	1.0e-2	7.2e-3	1.5e-3	9.0e12
P-XFEM	1.2e-2	3.2e-3	6.5e-3	2.7e9
D-XFEM	1.2e-2	6.9e-3	1.7e-3	6.2e12

Table 4: Discretization errors and matrix characteristics for Benchmark 1.

it provides the density of the associated matrix and its condition number for each method.

4.2 Benchmark 2: Regular Fracture Network

This test case is based on an article presenting a new dual continuum model, [17], with slightly modified boundary conditions and soil properties. The computational domain including the fracture network and boundary conditions are shown in Figure 10. The matrix permeability is set to $\boldsymbol{K}_{\mathrm{m}} = \mathbf{I}$, all fractures

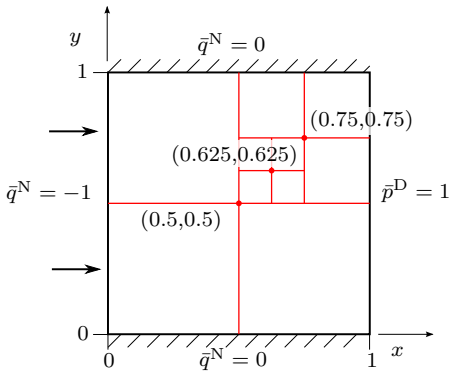


Fig. 10: Benchmark 2: Domain and boundary conditions.

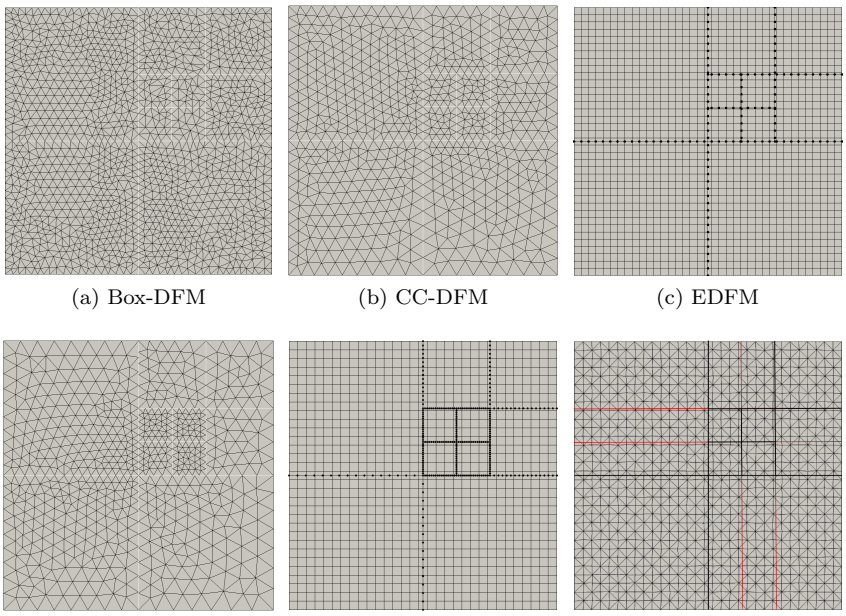
have a uniform aperture $d = 10^{-4}$. For the fracture permeablity we consider two cases: a highly conductive network with $k_{\mathrm{f,n}} = k_{\mathrm{f,t}} = 10^4$, as worked out in Section 4.2.1, and a blocking one with $k_{\mathrm{f,n}} = k_{\mathrm{f,t}} = 10^{-4}$, as described in Section 4.2.2 The reference solutions are computed on a grid which resolves every fracture with 10 elements in normal direction and becomes coarser away from the fractures. It has a total of 1,175,056 elements.

Table 5 lists the number of degrees of freedom, matrix elements and fracture elements for all the participating methods. The corresponding grids are

method	**d.o.f.**	**matrix elems**	**fracture elems**
Box-DFM	1422	2691 triangles	130
CC-DFM	1481	1386 triangles	95
EDFM	1501	1369 quads	132
Mortar-DFM	3366	1280 triangles	75
P-XFEM	1632	961 quads	318
D-XFEM	4474	1250 triangles	126
MFD			

Table 5: Grids for Benchmark 2.

visualized in Figure 11.



(a) Box-DFM (b) CC-DFM (c) EDFM

(d) Mortar-DFM (e) P-XFEM (f) D-XFEM

Fig. 11: Benchmark 2: the grids used by the different methods.

4.2.1 Conductive Fracture Network

First, we consider a highly conductive network by setting $k_{\mathrm{f,n}} = k_{\mathrm{f,t}} = 10^4$. The pressure distribution of the corresponding reference solution is shown in Figure 12.

The results of the different methods are first compared along two lines, one horizontal and one vertical, in Figure 13.

Table 6 lists the discretization errors for the different methods, particularly, the error for the matrix domain and the one along the fracture network.

method	**matrix error**	**fracture error**	**nnz/size^2**	**$\|\cdot\|_2$-condition**
Box-DFM	6.7e-3	1.1e-3	4.7e-3	7.9e3
CC-DFM	1.1e-2	5.0e-3	2.7e-3	5.6e4
EDFM	6.5e-3	4.0e-3	3.3e-3	5.6e4
Mortar-DFM	1.0e-2	7.4e-3	1.8e-3	2.4e6
P-XFEM	1.7e-2	6.0e-3	7.8e-3	6.8e9
D-XFEM	9.6e-3	8.9e-3	1.3e-3	1.2e6

Table 6: Discretization errors and matrix characteristics for Benchmark 2.

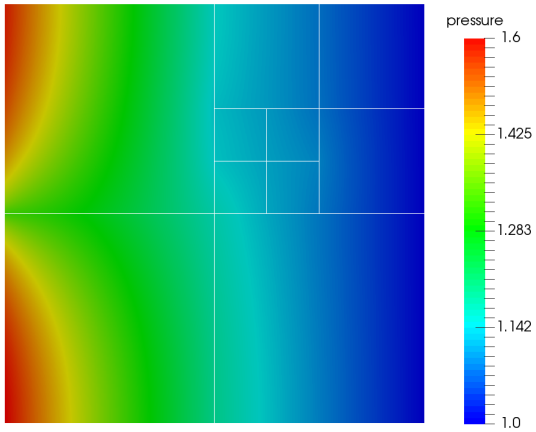
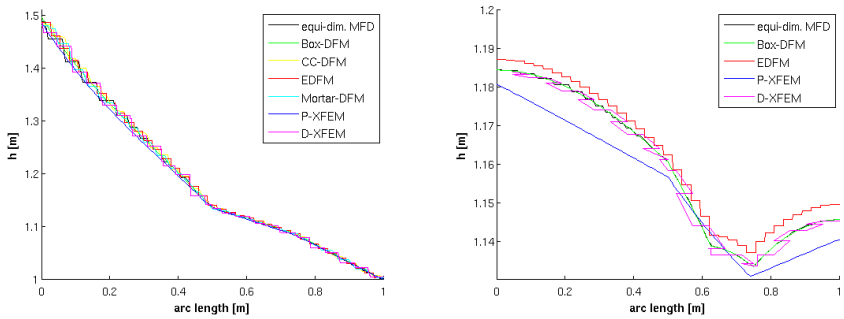


Fig. 12: Benchmark 2 with conductive fractures: pressure reference solution.



(a) Horizontal line at $y = 0.7$. (b) Longest vertical fracture at $x = 0.5$.

Fig. 13: Benchmark 2: comparison of values along two lines.

Moreover, it provides the density of the associated matrix and its condition number for each method.

The results for the conducting fractures are similar to those presented for the first benchmark. In particular, the performance of the methods is comparable as shown by both the matrix and the fracture errors. In fact, since the degree of sparsity does not differ significantly either, the only notable differences between the methods are the number of degrees of freedom and the condition numbers. In that context, the Mortar-DFM and D-XFEM are the clear outliers, containing a large number of degrees of freedom due to the incorporated flux variable and resulting in high condition numbers. The P-XFEM scheme exhibits the highest condition number yet it has much fewer degrees of freedom.

4.2.2 Blocking Fracture Network

We now assume a blocking fracture network by setting $k_{\mathrm{f,n}} = k_{\mathrm{f,t}} = 10^{-4}$. The pressure distribution of the corresponding reference solution is shown in Figure 14.

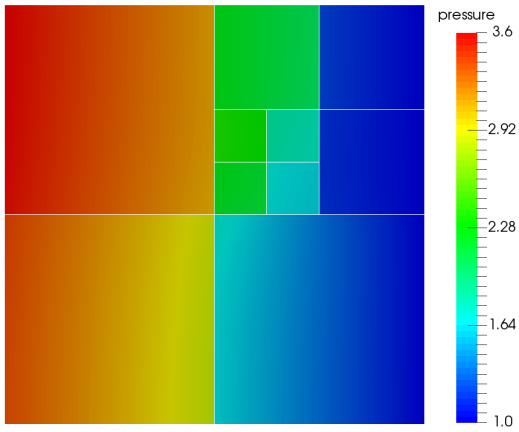


Fig. 14: Benchmark 2 with blocking fractures: pressure reference solution.

Figure 15 compares the results of the different methods along a diagonal line throughout the whole domain from $(0.0, 0.1) - (0.9, 1.0)$.

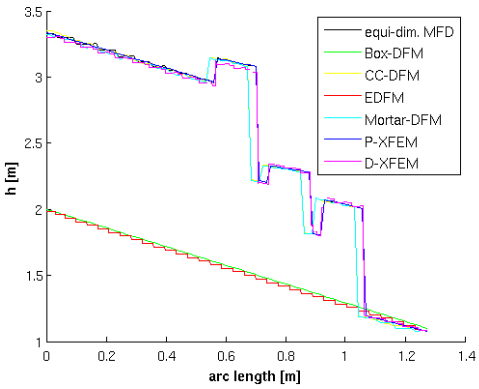


Fig. 15: Benchmark 2: values along the line $(0.0, 0.1) - (0.9, 1.0)$.

Table 7 lists the discretization errors for the different methods, particularly, the error for the matrix domain and the one along the fracture network.

method	**matrix error**	**fracture error**	**nnz/size^2**	**$\|\cdot\|_2$-condition**
Box-DFM	4.1e-1	3.8e-1	4.7e-3	3.5e3
CC-DFM	5.7e-3	4.6e-3	2.7e-3	2.6e4
EDFM	2.9e-1	3.2e-1	3.3e-3	9.2e3
Mortar-DFM	4.5e-3	4.9e-3	1.6e-3	9.0e2
P-XFEM	2.9e-3	2.2e-2	8.1e-3	2.0e4
D-XFEM	1.0e-2	1.9e-2	1.3e-3	2.2e6

Table 7: Discretization errors and matrix characteristics for Benchmark 2.

Moreover, it provides the density of the associated matrix and its condition number for each method.

With blocking fractures, the distinction between the different methods is more apparent. As mentioned above, the Box-DFM and EDFM schemes are unable to capture the resulting pressure discontinuities. As a result, these methods show large errors in both the matrix and the fracture domains. The remaining methods, which are capable of handling discontinuities, do not differ significantly among each other in terms of fracture and matrix errors. We do note that the condition numbers have improved significantly for the Mortar-DFM and P-XFEM schemes. For CC-DFM and D-XFEM, condition numbers for the situation with fractures blocking the flow are similar to those obtained for the situation where the fractures were conduits for flow.

4.3 Benchmark 3: Complex Fracture Network

This testcase considers a small, but complex fracture network tht includes permeable and blocking fractures. The domain and boundary conditions are shown in Figure 16. The exact coordinates for the fracture positions are listed in Table 8. The fracture network contains ten straight immersed fractures, grouped in disconnected networks. The aperture is $d = 10^{-4}$ for all fractures, and permeability is $k_{\mathrm{f,n}} = k_{\mathrm{f,t}} = 10^4$ for al fractures except for fractures 4 and 5 which are blocking fractures with $k_{\mathrm{f,n}} = k_{\mathrm{f,t}} = 10^{-4}$. Note that we are considering two subcases a) and b) with a pressure gradient which is predominantly vertical and horizontal respectively, to better highlight the impact of the blocking fractures.

Table 8: Benchmark 3: Domain coordinates

Nf	xA	yA	xB	yB
1	0.0500	0.4160	0.2200	0.0624
2	0.0500	0.2750	0.2500	0.1350
3	0.1500	0.6300	0.4500	0.0900
4	0.1500	0.9167	0.4000	0.5000
5	0.6500	0.8333	0.8500	0.1667
6	0.7000	0.2350	0.8500	0.1675
7	0.6000	0.3800	0.8500	0.2675
8	0.3500	0.9714	0.8000	0.7143
9	0.7500	0.9574	0.9500	0.8155
10	0.1500	0.8363	0.4000	0.9727

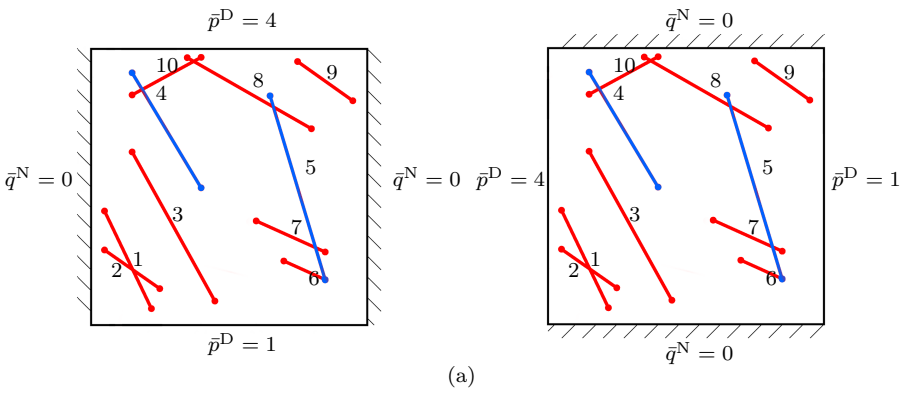
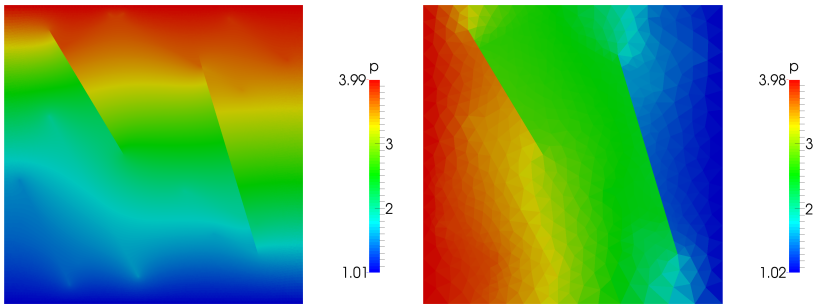


Fig. 16: Benchmark 3: Domain and boundary conditions for cases a) and b)

Table 9 lists the number of degrees of freedom, matrix elements and fracture elements for all the participating methods. The corresponding grids are visualized in Figure 18.



(a) (b)

Fig. 17: Benchmark 3: reference solution for cases a) and b)

method	**d.o.f.**	**matrix elems**	**fracture elems**
Box-DFM	1460	2838 triangles	155
CC-DFM	1510	1407 triangles	103
EDFM	1572	1369 quads	203
Mortar-DFM	3953	1452 triangles	105
P-XFEM	2028	1122 quads	376
D-XFEM	7180	1922 triangles	199
MFD			

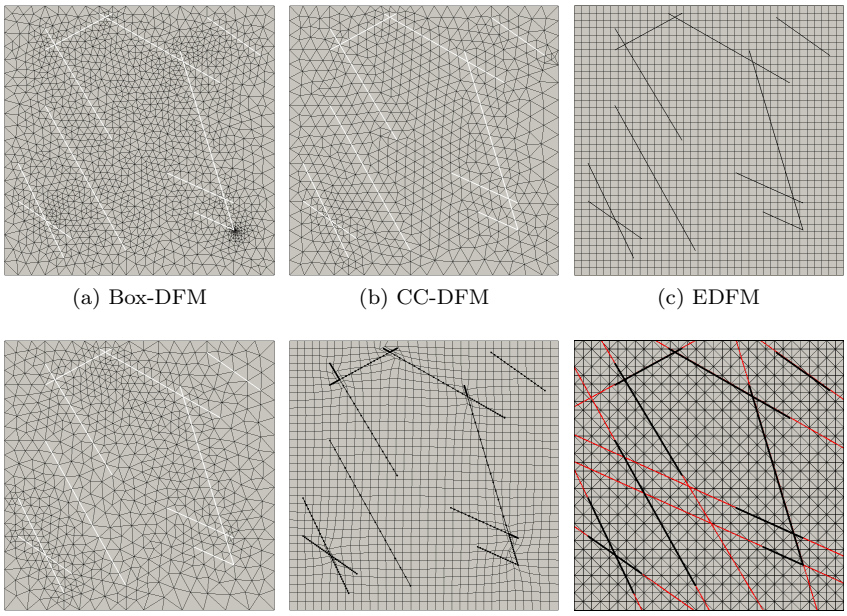
Table 9: Grids for Benchmark 3.

4.3.1 Flow from top to bottom

Table 10 lists the discretization errors for the first variant, namely, the flow from top to bottom.

method	**matrix error**	**fracture error**	**nnz/size^2**	**$\|\cdot\|_2$-condition**
Box-DFM	4.4e-2	3.8e-2	4.6e-3	4.5e3
CC-DFM	2.6e-2	3.3e-2	2.7e-3	3.8e4
EDFM	3.8e-2	4.5e-2	3.1e-3	1.2e6
Mortar-DFM	1.0e-2	1.7e-2	1.4e-3	1.1e6
P-XFEM	9.9e-2	1.2e-1	6.2e-3	2.0e15
D-XFEM	1.9e-2	2.9e-2	8.2e-4	8.1e3

Table 10: Discretization errors and matrix characteristics for the first variant of Benchmark 3.



(a) Box-DFM (b) CC-DFM (c) EDFM

(d) Mortar-DFM (e) P-XFEM (f) D-XFEM

Fig. 18: Benchmark 3: the grids used by the different methods.

4.3.2 Flow from left to right

Table 11 lists the discretization errors for the second variant, namely, the flow from left to right.

method	**matrix error**	**fracture error**	**nnz/size^2**	**$\|\cdot\|_2$-condition**
Box-DFM	7.5e-2	7.0e-2	4.6e-3	5.6e3
CC-DFM	5.2e-2	7.3e-2	2.7e-3	4.5e4
CC-DFM*	1.1e-2	2.7e-2	2.6e-3	8.1e5
EDFM	5.8e-2	8.9e-2	3.1e-3	1.2e6
Mortar-DFM	1.3e-2	2.7e-2	1.4e-3	7.3e8
P-XFEM				
D-XFEM	2.2e-2	3.6e-2	8.2e-4	8.1e3

Table 11: Discretization errors and matrix characteristics for the second variant of Benchmark 3.

Even if this is still a synthetic case the geometry of the network starts to be an issue: relatively small intersection angles are present, for instance, between fractures 1 and 2. Another difficulty consists in the coexistence of permeable

and blocking fracture which intersect each other: on one hand, some of the methods are not well suited to describe a blocking behavior, on the other hand the coupling conditions at the intersection become less trivial in these cases. All the participating methods that account explicitly for the effect of permeability at the fracture intersections have adopted the harmonic average in the case of a permeable and a blocking fracture crossing each other.

The errors reported in Table 10 show that the methods requiring the continuity of pressure (EDFM and the Box-DFM) exhibit slightly higher errors in the matrix. However, the difference is not particularly sharp, since in the first sub-case the average pressure gradient is almost parallel to the blocking fractures. In the case b), where we impose pressure on the sides of the square domain, the solution is more challenging as we can observe from Figure 17 and the gap between continuous and discontinuous methods increases. However, it should be noted that the errors remain of the same order of magnitude, indicating that all the methods capture the overall trend of the solution. The elimination of the fracture intersection cells in the CC-DFM is ill-suited for cases where fractures of different permeability cross, as the information about the intersection permeability is lost. Therefore, we include a solution CC-DFM* for which we have not performed the removal for case b). The 11 values show a far better result compared to the CC-DFM with elimination, but also demonstrate that the elimination significantly reduces the condition number.

4.4 Benchmark 4: a real case

In this last test case we consider a real set of fractures from an interpreted outcrop in the Sotra island, out of Bergen in Norway. The set is composed of 64 fractures grouped in 13 different connected networks, ranging from isolated fractures up to tens of fractures each. In the interpretation process two fractures were composed by more than one segment however, to easier the process, we substitute them by a single line. It is worth to notice that we are changing the connectivity of the system, nevertheless our goal is to make a comparison of the previous schemes on a complex set of fractures. The interpreted outcrop and the corresponding set of fractures are represented in Figure 19. The size of the rock matrix is $700m \times 600m$ with uniform scalar permeability equal to $10^{-14}m^2$. For simplicity all the fractures have the same scalar permeability equal in the tangential and normal direction, $10^{-8}m^2$, and aperture $10^{-2}m$. We consider no-flow boundary condition on top and bottom, pressure $1013250Pa$ on the left, and pressure $0Pa$ on the right of the boundary of the domain. Due to the complexity of the network only some of the numerical schemes can be used, nevertheless it is worth to point out that for the others the main difficulty is to enhance the code and not the scheme to handle such geometry.

Table 12 lists the number of degrees of freedom, the density of the associated matrix, and its condition number for the different methods. Due to the

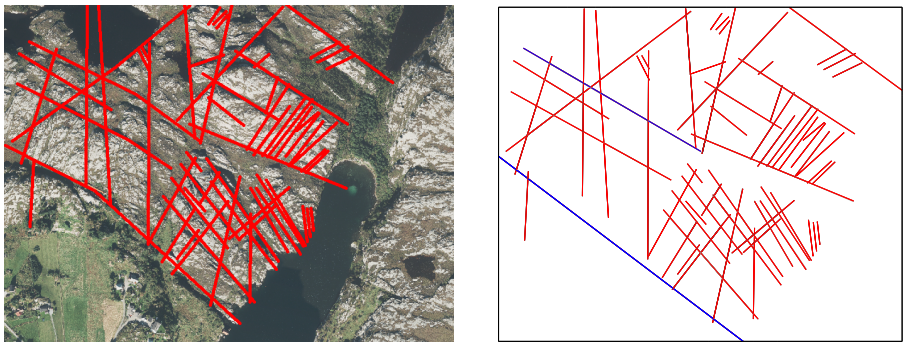


Fig. 19: In the left the interpretation of the set of fractures superimposed to the map. In the right the geometry used in the simulations. The rectified fractures are depicted in blue.

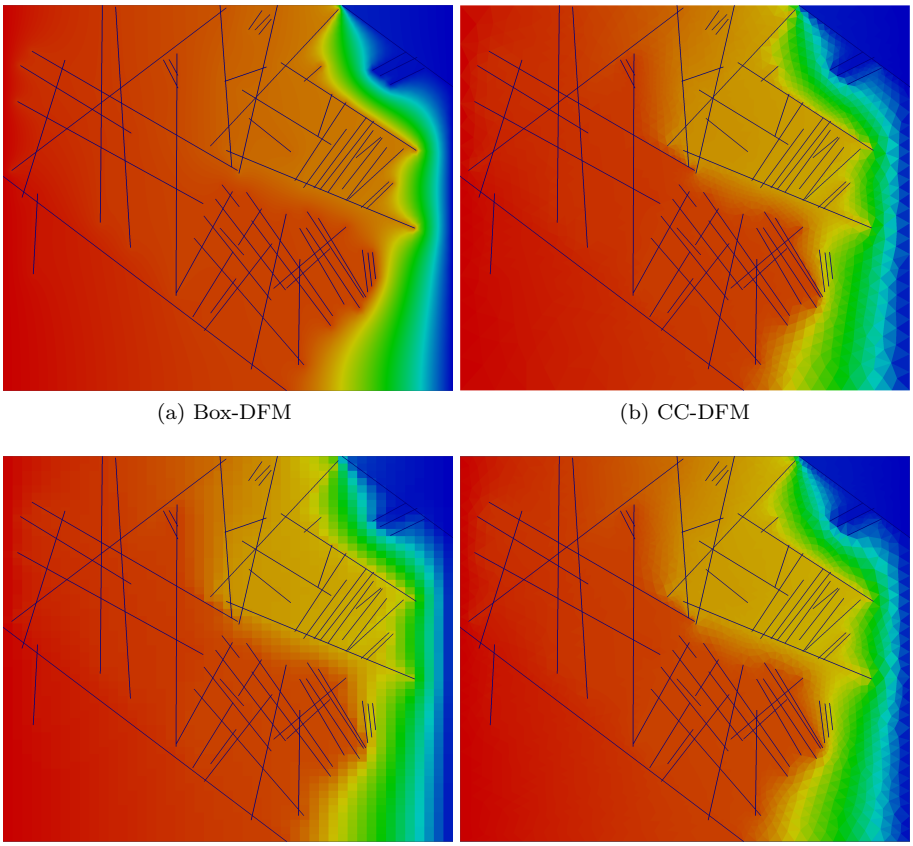
method	**d.o.f.**	**matrix elems**	**fracture elems**	**nnz/size^2**	**$\|\cdot\|_2$-cond.**
Box-DFM	5563	10807 triangles	1386	1.2e-3	9.3e5
CC-DFM	8481	7614 triangles	867	4.9e-4	5.3e6
EDFM	3599	2491 quads	1108	1.4e-3	4.7e6
Mortar-DFM	25258	8319 triangles	1317	2.0e-4	2.2e17

Table 12: Discretization and matrix characteristics for Benchmark 4.

geometrical difficulties of the network the request of having a similar number of degrees of freedom among the methods is relaxed, as Table 12 indicates. Considering Figure 22, the solutions are reported for the four methods. We notice that, except for the top right part of the domain in the Box-DFM method, the solutions are similar and comparable which is an indication of their correctness. Compare to the previous test cases the mesh generation is the main concern and some of the methods require a fine tuning to avoid unphysical connections among elements where the fracture are close. An example can be found in the middle of the domain and reported in Figure 21. Only EDFM is more robust with respect to this constraint. To present a more detailed comparison among the methods, Figure 22 represents the pressure solution along two different lines: for $y = 500m$ and for $x = 625m$. We can notice that the methods behave similarly, and the Box-DFM slightly overestimates some picks. The oscillation of the methods are related to mesh effects.

5 Summary and Conclusion

To simulate fractured porous media systems, you have to make several choices. The first question to answer is, if you need explicit fracture representations. Within this manuscript, we presented only discrete fracture models by means of several examples and corresponding solutions of different methods and nu-

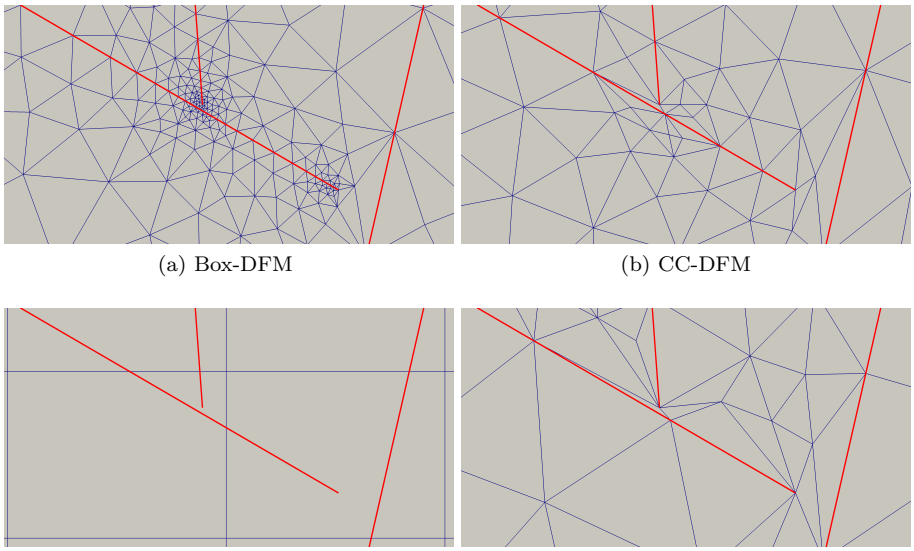


(a) Box-DFM (b) CC-DFM

(c) EDFM (d) Mortar-DFM

Fig. 20: Representation of the matrix pressures field for the realistic case. The solutions have range in $[0, 101325]Pa$, a "Blue to Red Rainbow" colour map is used.

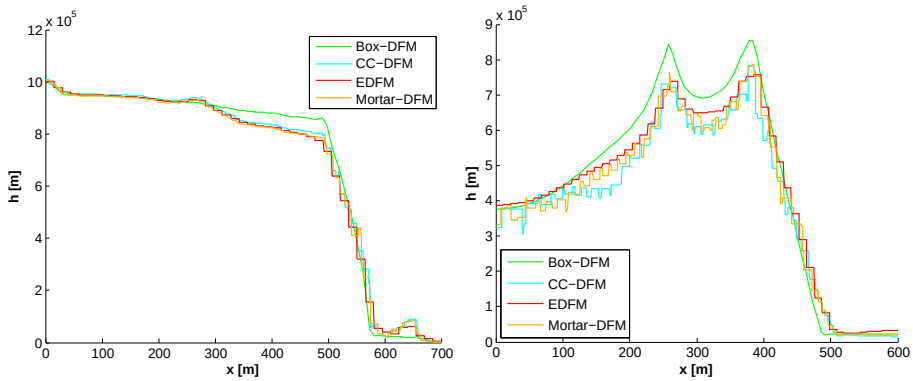
merical methods. For highly conductive fractures and simple-to-grid fracture networks, the DFM model shows its strengths. The XFEM method can match the reference solution, but in some cases it needs more degrees of freedom than the DFM to achieve comparable results. The advantage of independent fracture and matrix grids is clearly visible and leads in most cases to a huge advantage over a uniform refinement. The presented XFEM method is suited best for flow regimes depending on both, the fractures and the surrounding rock matrix and is able to account for highly conductive but also almost impermeable fractures with respect to the surrounding matrix. For fractures with a lower permeability than the matrix, the DFM model is not able to capture the flow field correctly and the XFEM method shows very accurate results on relatively coarse grids.



(a) Box-DFM (b) CC-DFM

(c) EDFM (d) Mortar-DFM

Fig. 21: Representation of mesh in the middle of the domain. The size of the picture is approximately $30m \times 15m$ centered in $(360, 350)$.



(a) $y = 500$ (b) $x = 625$

Fig. 22: Plot over line of the pressure solution for the methods.

6 Acknowledgements

The authors warmly thank Luisa F. Zuluaga, from University of Bergen, for constructing and providing the real fracture network for the example in Section 4.4. The authors wish to thanks also Luca Pasquale and Stefano Zonca.

The fourth author acknowledges financial support for the ANIGMA project from the Research Council of Norway (project no. 244129/E20) through the ENERGIX program.

References

1. A. Assteerawatt. *Flow and transport modelling of fractured aquifers based on a geostatistical approach.* PhD thesis, Universität Stuttgart, 2008.
2. P. Bastian, M. Blatt, A. Dedner, C. Engwer, R. Klöfkorn, R. Kornhuber, M. Ohlberger, and O. Sander. A generic grid interface for parallel and adaptive scientific computing. Part II: Implementation and tests in DUNE. *Computing*, 82(2-3):121–138, 2008.
3. P. Bastian, F. Heimann, and S. Marnach. Generic implementation of finite element methods in the distributed and unified numerics environment (DUNE). *Kybernetika*, 46(2):294–315, 2010.
4. B. Berkowitz. Characterizing flow and transport in fractured geological media: A review. *Advances in Water Resources*, 25(8-12):861–884, 2002.
5. F. Brezzi, K. Lipnikov, and V. Simoncini. A family of mimetic finite difference methods on polygonal and polyhedral meshes. *Math. Mod. Meth. Appl. S.*, 15(10):1533–1551, 2005.
6. C. D'Angelo and A. Scotti. A mixed finite element method for darcy flow in fractured porous media with non-matching grids. *ESAIM: Mathematical Modelling and Numerical Analysis*, 46(02):465–489, 2012.
7. J. S. de Araujo Cavalcante Filho, M. Shakiba, A. Moinfar, and K. Sepehrnoori. Implementation of a preprocessor for embedded discrete fracture modeling in an IMPEC compositional reservoir simulator. In *SPE Reservoir Simulation Symposium, 23-25 February, Houston, Texas, USA*. Society of Petroleum Engineers, 2015.
8. P. Dietrich, R. Helmig, M. Sauter, H. Hötzl, J. Köngeter, and G. Teutsch. *Flow and transport in fractured porous media.* Springer, 2005.
9. R. Eymard, T. Gallouët, and R. Herbin. Finite volume methods. In P. G. Ciarlet and J. L. Lions, editors, *Solution of Equation in $\mathcal{R}^n$ (Part 3), Techniques of Scientific Computing (Part 3)*, volume 7 of *Handbook of Numerical Analysis*, pages 713–1018. Elsevier, 2000.
10. A. Firoozabadi and J. E. P. Monteagudo. Control-volume method for numerical simulation of two-phase immiscible flow in two- and three-dimensional discrete-fractured media. *Water Resources Research*, 40:W07405, July 2004.
11. B. Flemisch, M. Darcis, K. Erbertseder, B. Faigle, A. Lauser, K. Mosthaf, S. Müthing, P. Nuske, A. Tatomir, M. Wolff, and R. Helmig. DuMu^{x}: DUNE for multi-{phase, component, scale, physics, ...} flow and transport in porous media. *Advances in Water Resources*, 2011.
12. B. Flemisch, A. Fumagalli, and A. Scotti. A review of the XFEM-based approximation of flow in fractured porous media. 2016. Submitted.
13. B. Flemisch and R. Helmig. Numerical investigation of a mimetic finite difference method. In R. Eymard and J. Hérard, editors, *Finite Volumes for Complex Applications V – Problems and Perspectives*, pages 815–824. Wiley - VCH, 2008.
14. L. Formaggia, A. Fumagalli, A. Scotti, and P. Ruffo. A reduced model for Darcy's problem in networks of fractures. *MOX report 32*, 2012.
15. A. Fumagalli, L. Pasquale, S. Zonca, and S. Micheletti. An upscaling procedure for fractured reservoirs with non-matching grids. Technical Report 33/2015, Politecnico di Milano, 2015.
16. A. Fumagalli and A. Scotti. An efficient XFEM approximation of Darcy flows in fractured porous media. *Oil and Gas Sciences and Technologies - Revue d?IFP Energies Nouvelles*, 69.4:555–564, 2014.
17. S. Geiger, M. Dentz, and I. Neuweiler. A novel multi-rate dual-porosity model for improved simulation of fractured and multi-porosity reservoirs. In *SPE Reservoir Characterisation and Simulation Conference and Exhibition*, 2011.

18. H. Hajibeygi, D. Karvounis, and P. Jenny. A hierarchical fracture model for the iterative multiscale finite volume method. *Journal of Computational Physics*, 230(24):8729–8743, 2011.
19. A. Hansbo and P. Hansbo. An unfitted finite element method, based on Nitsche's method, for elliptic interface problems. *Computer methods in applied mechanics and engineering*, 191(47-48):5537–5552, 2002.
20. R. Helmig. *Multiphase Flow and Transport Processes in the Subsurface: A Contribution to the Modeling of Hydrosystems*. Springer, 1 edition, Oct. 1997.
21. H. Hoteit and A. Firoozabadi. Numerical modeling of two-phase flow in heterogeneous permeable media with different capillarity pressures. *Advances in Water Resources*, 31(1):56–73, 2008.
22. M. Karimi-Fard, L. J. Durlofsky, and K. Aziz. An efficient discrete-fracture model applicable for general-purpose reservoir simulators. *SPE Journal*, 9(2):227–236, 2004.
23. L. Li and S. H. Lee. Efficient field-scale simulation of black oil in a naturally fractured reservoir through discrete fracture networks and homogenized media. *SPE Reservoir Evaluation & Engineering*, 11:750–758, 2008.
24. V. Martin, J. Jaffré, and J. E. Roberts. Modeling fractures and barriers as interfaces for flow in porous media. *SIAM Journal on Scientific Computing*, 26(5):1667–1691, 2005.
25. S. K. Matthäi and M. Belayneh. Fluid flow partitioning between fractures and a permeable rock matrix. *Geophysical Research Letters*, 31(7):7602–6, 2004.
26. A. Moinfar, W. Narr, M.-H. Hui, B. T. Mallison, and S. H. Lee. Comparison of discrete-fracture and dual-permeability models for multiphase flow in naturally fractured reservoirs. In *SPE Reservoir Simulation Symposium*. Society of Petroleum Engineers, 2011.
27. A. Moinfar, A. Varavei, K. Sepehrnoori, and R. T. Johns. Development of an efficient embedded discrete fracture model for 3d compositional reservoir simulation in fractured reservoirs. *SPE Journal*, 19(02):289 – 303, 2014.
28. A. Moinfar, A. Varavei, K. Sepehrnoori, and R. T. Johns. Development of an efficient embedded discrete fracture model for 3d compositional reservoir simulation in fractured reservoirs. *Society of Petroleum Engineers*, 19(2):289–303, April 2014.
29. S. Müthing. *A flexible framework for multi physics and multi domain PDE simulations*. PhD thesis, Universität Stuttgart, 2015.
30. S. P. Neumann. Trends, prospects and challenges in quantifying flow and transport through fractured rocks. *Hydrogeol Journal*, 13:124–147, 2005.
31. P. Panfili and A. Cominelli. Simulation of miscible gas injection in a fractured carbonate reservoir using an embedded discrete fracture model. In *Abu Dhabi International Petroleum Exhibition and Conference, 10-13 November, Abu Dhabi, UAE*. Society of Petroleum Engineers, 2014.
32. P. Panfili, A. Cominelli, and A. Scotti. Using Embedded Discrete Fracture Models (EDFMs) to Simulate Realistic Fluid Flow Problems. In *Second EAGE Workshop on Naturally Fractured Reservoirs, Muscat, Oman*, 2013.
33. V. Reichenberger, H. Jakobs, P. Bastian, and R. Helmig. A mixed-dimensional finite volume method for two-phase flow in fractured porous media. *Advances in Water Resources*, 29(7):1020–1036, July 2006.
34. M. Sahimi. *Flow and transport in porous media and fractured rock: from classical methods to modern approaches*. John Wiley & Sons, 2011.
35. T. H. Sandve, I. Berre, and J. M. Nordbotten. An efficient multi-point flux approximation method for discrete fracture–matrix simulations. *Journal of Computational Physics*, 2012.
36. N. Schwenck. *An XFEM-Based Model for Fluid Flow in Fractured Porous Media*. PhD thesis, University of Stuttgart, Department of Hydromechanics and Modelling of Hydrosystems, 2015.
37. N. Schwenck, B. Flemisch, R. Helmig, and B. Wohlmuth. Dimensionally reduced flow models in fractured porous media: crossings and boundaries. *Computational Geosciences*, 19(6):1219–1230, 2015.
38. B. B. S. Singhal and R. P. Gupta. *Applied hydrogeology of fractured rocks*. Springer Science & Business Media, 2010.
39. Swedish Nuclear Power Inspectorate (SKI). The international hydrocoin project–background and results. *Paris, France: Organization for Economic Co-operation and Development*, 1987.

40. A.-B. Tatomir. *From discrete to continuum concepts of flow in fractured porous media.* PhD thesis, Universität Stuttgart, Holzgartenstr. 16, 70174 Stuttgart, 2012.
41. M. Tene, M. S. Al Kobaisi, and H. Hajibeygi. Multiscale projection-based Embedded Discrete Fracture Modeling approach (F-AMS-pEDFM). In *ECMOR XIV-15th European Conference on the Mathematics of Oil Recovery.* EAGE, 2016.