Heat Transfer Upscaling in Geothermal Reservoirs

Master of Science Thesis
in Applied and Computational Mathematics

Anna Varzina

Department of Mathematics
University of Bergen

November 20, 2015

Acknowledgements

I would like to thank my supervisor Inga Berre for your inestimable guidance, support and valuable feedback. I am deeply grateful to my second supervisor Anna Nissen for your contribute to my studying. I appreciate your generous help and time you spend sharing with me your experience and I learned a lot while discussing with you progress of the work and accurate writing. Sincere gratitude to the Department of Mathematics and the NUPUS cooperation of universities and scientists for inspiration from unforgettable student trip to the United States.

I thank all my friends in Bergen, especially Anna Kvashchuk, who experienced almost the same emotions while studying in the University of Bergen. Finally, thanks to my parents, grandparents, fiancé and friends in Russia, who were always with me mentally. I am much obliged to you for my confidence and enthusiasm while writing this thesis.

Anna,
November 2015.

Abstract

In this thesis we implement a numerical model of heat transfer in geothermal reservoirs. We use existing pressure and flow transport solvers as a starting point to investigate discretization techniques for a convection-conduction temperature equation. Then we develop and analyse two different heat transfer solvers: explicit and implicit, that have different accuracy and convergence requirements. For the convective part of the energy equation the upwind scheme is implemented and the two-point flux approximation is used to discretize the conductive term. Usually heat transfer simulations require large computational time due to high resolution on a fine scale. For efficient computation we investigate flow-based upgridding techniques, which were used before for fluid transport in porous media. However upgridding and upscaling can lead to less accurate results due to much loss of details in a discrete model. We compare solutions on different types of grids such as Cartesian grid and flow-based grids that are generated according to various indicators like permeability, velocity, time-of-flight and thermal conductivity. In this work we simulate an initial-boundary value problem with a heat flow through boundaries and try to investigate, which coarse grid leads to the most accurate results when solving the energy equation.

Contents

Contents v

1 Introduction **1**

1.1 Geothermal energy 3

2 Fluid Flow in Porous Media **5**

2.1 Rock and fluid parameters 6

2.1.1 Porosity 6

2.1.2 Permeability 6

2.1.3 Density and viscosity 7

2.2 Darcy's Law 8

2.3 Conservation of Mass 9

2.3.1 Boundary Conditions 11

2.3.2 Well model 12

2.4 Time of Flight 13

3 Heat transfer **15**

3.1 Conduction 16

3.2 Convection 17

3.3 The conservation of energy 18

3.3.1 Initial and boundary conditions 20

3.4 Dimensionless numbers 21

4 Numerical Methods **23**

4.1 Discretization in space 23

4.1.1 Two-Point Flux Approximation 24

4.1.2 Upwind scheme 26

4.2 Discretization in Time 27

4.3 Implementation of Boundary and Initial conditions 29

4.4 Convergence 32

5 Coarse Scale Methods **33**
5.1 Upscaling 33
5.2 Upgridding 36

6 Numerical Experiments **39**
6.1 Fluid Flow Model 40
6.1.1 Homogeneous medium 40
6.1.2 10th SPE Model 41
6.2 Heat Transport Model 42
6.2.1 Boundary flux problem in a homogeneous medium 42
6.2.2 Boundary flux problem in a heterogeneous medium 44
6.2.3 Quarter-five spot problem and numerical diffusion. 47
6.3 Numerical Experiments on a Coarse Scale 50
6.3.1 Homogeneous medium 50
6.3.2 Heterogeneous medium 52
6.3.3 Channeled heterogeneous medium 56

7 Conclusion **61**

List of Symbols **63**

List of Figures **65**

List of Tables **67**

Bibliography **69**

Chapter 1

Introduction

This thesis is devoted to the simulation of heat transfer in geothermal reservoirs. Energy consumption grows with each passing day due to the technological development and the increase of the world's population. Thus the improvement of the energy supply becomes critical even now. Renewable energy sources become more popular and nowadays this type of energy contributes 22.8% of global electricity supply according to REN21's 2015 report [30]. Geothermal energy is the renewable energy source and this work deals with numerical simulations for geothermal energy production, which will be discussed in the following section.

The motivation for studying of heat transfer is related to geothermal energy extraction. The mathematical modelling is important part of research, it helps to get a better understanding of the process and develop efficient ways of the heat extraction. Usual heat transfer simulations are very large and have too much details to be represented in a discrete model. The typical size of a simulation grid is $10^5 - 10^8$ grid cells and certainly not every reservoir simulators have enough computational capacity to process such detailed models. Our goal is to reduce the computational time and at the same time maintain good accuracy. There are two ways to reach this goal: to develop effective numerical methods or to implement profitable upscaling techniques, that hopefully will capture the most important features in simulations. In this work to make our simulations work faster we create a coarse grid using amalgamation of the fine grid cells [17], [16]. We combine discretization strategies on both fine and coarse grids to balance acuracy and computational speed. We first obtain fine scale pressure and flux values and apply these parameters to solve the energy conservation equation. In our simulations we investigate different coarse scale partitions and analyse which grid can be the most efficient for our test case.

In this chapter we will explain the concept of geothermal energy extraction and briefly describe the structure of the thesis.

In **Chapter 2** we present some definitions and formulas from fluid mechanics and reservoir engineering that are necessary for our research. The governing equations of the flow in a porous medium are also introduced in this chapter that are absolutely essential for heat transfer model in geothermal reservoirs.

In **Chapter 3** the heat transport is discussed. We introduce necessary definitions and properties of a heat flow, which are used in the construction of energy equation. We explain different heat processes and gradually derive the energy conservation equation for temperature. We apply the heat transfer to the porous media and discuss boundary and initial conditions.

In **Chapter 4** the numerical methods are presented, which are used to solve partial differential equations from Chapter 2 and 3. We discretize the pressure equation and derive explicit and implicit schemes for the energy conservation equation. The implementation of boundary conditions and sources in numerical schemes is explained here. Also we discuss temporal and spatial convergence of the implemented numerical methods.

In **Chapter 5** we introduce the coarse scale methods. Different upscaling techniques, that are applicable to various parameters in our model, are explained here in details. Also we consider two types of upgridding procedures that we use in the simulation of heat transfer equation. Much attention is given to the flow-based nonuniform coarsening algorithm, which was developed to generate coarse grids for fluid transport simulation.

In **Chapter 6** the numerical experiments are presented and discussed. We implement the pressure equation and the conservation of energy equation on homogeneous and heterogeneous media in the MATLAB Reservoir Simulation Toolbox (MRST). We simulate boundary heat flux problem and compare solutions obtained with two solvers that are based on different discretization schemes for Euler's forward method and Euler's backward method. Then we simulate numerical experiments on different coarse grids for both homogeneous and heterogeneous media and compare the accuracy error at various Péclet numbers.

In **Chapter 7** we discuss the obtained results and give the conclusion of this work.

1.1 Geothermal energy

Geothermal energy is thermal energy that is generated and contained in the Earth crust interior. It is usually described as an alternative energy source (not fossil fuels) and it a type of renewable energy resource which we can use almost independently from weather conditions, as distinct from wind or solar energy sources. Many countries like the United States, the Philippines, Indonesia, Mexico, New Zealand, Italy, Iceland and others make active use of geothermal energy [30].

Geothermal energy has many advantages and first of all its reneawability and continuity. Due to sustainability of this type of energy we can save other energy resources for future generations. However geothermal energy extraction can have significant influence on the environment. Gases like carbon dioxide (CO_2), hydrogen sulfide (H_2S) and others may be present in the deep ground. These greenhouse gases affect the global warming, but the amount of such gases emitted by geothermal plants is much less compared to emission intensity of fossil fuel plants. Also ground waters can contain some toxic elements and we should be careful with the refinery procedures and reduce the risk of environmental pollution. Certainly the disruption of nature due to the geothermal energy extraction is much less in comparison with fossil fuels. Another minus is the investment cost of the drilling. The diameter of the wells for geothermal purposes is larger than oil and gas wells' average diameter and thus it is cost-intensive to drill such wells. On a positive note geothermal plants are comparable to other energy sources over a long term because the operation cost is low and the price of geothermal energy per kilowatt-hour is lower than for many other energy types [23].

Thermal regions are available in many parts of the world and of course the best place for the construction of geothermal power plants is the area around the edges of continental plates, where the crust is thinner. Geothermal energy is a large resource that can provide both electrical power and heat [35]. Earth allocates about $44 \cdot 10^{12}$ W of heat [28], of which only 2% is absorbed in the Earth crust. Approximately the amount of geothermal power capacity reached 12.8 GW in 2015 and it is expected to reach more than 14.5 GW by 2020 [14]. According to various estimates the temperature at the core of the Earth is at least 6650 ℃ and the geothermal gradient in the well rises by 25 ℃ every km of depth in avarage (from 15 ℃/km to 50 ℃/km) [35], [12]. This heat is delivered to the surface in the form of steam or hot water and it can be used directly to heat houses and to produce electricity. For instance, in enhanced geothermal system (EGS) cold water is injected at high pressure into reservoir. The water is then heated by the earth crust interior and returned back to the surface through production

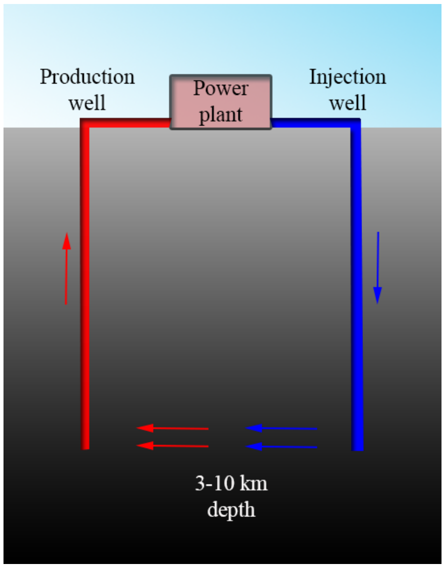


Figure 1.1: Enhanced geothermal system.

wells (see Figure 1.1). The heat is extracted and turned into electricity by power plants. When the fluid is cooled the process is repeated.

The geothermal reservoir represents a porous rock at high temperatures, where the water is transferred through. The heat conducts from the rock to the fluid (or vapour) and then it is transported with water by convection and conduction. In the following chapters we will consider in details the processes of the fluid flow and the heat transfer through the porous material.

Chapter 2

Fluid Flow in Porous Media

To investigate the heat transport in geothermal reservoirs it is important to introduce some basic equations and definitions of a fluid transport in a porous medium. A *porous medium* is a material that contains some channels and isolated pores (void) where the fluid can be present. The solid part of this material is referred to as a skeleton or a matrix (Figure 2.1). Usually the structure and the shape of the medium is not known and it is commonly assumed that the porous medium is a continuum with some macroscopic parameters. We consider such properties of the porous medium over particular length scale which is referred to as *representative elementary volume* (REV). The typical size of REV in soil is between 1 cm^3 and 1 dm^3, so it does not depend on the micro-scale structure of individual pores and excludes large scale heterogeneities [4], [26], [29].

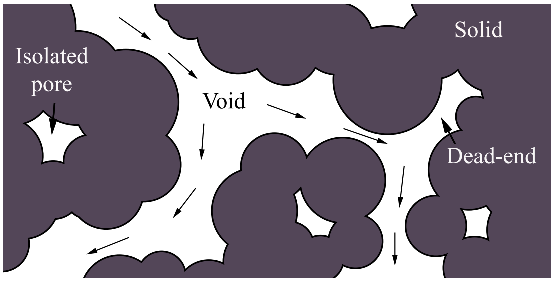


Figure 2.1: Porous Medium.

For a single-phase flow pores are filled with just one fluid and for a multiphase flow pores are filled with two or more fluids. In this work we consider that we have a reservoir filled with one fluid and below we are going to describe porous media parameters and governing equations related to a single-phase flow. These parameters are called petrophysical properties which are neither geological nor geophysical but engineering quantities, that are used as input to flow simulators.

2.1 Rock and fluid parameters

2.1.1 Porosity

The simplest way to characterize the pore space is a dimensionless parameter that is called *porosity*, which can be found as the ratio of the total pore volume in REV to the total volume of the REV [26]. The porosity, usually denoted by ϕ, is defined by

$$\phi = \frac{\text{volume of void in REV}}{\text{volume of REV}}.$$

So we can see that porosity is a dimensionless value in the $[0, 1]$ interval. Also the void space consists of two parts: the channels or interconnected pore space and isolated pores and dead-end pores that are unavailable for the fluid flow. That is why when the porosity is mentioned it means that only the connected part of the medium is considered and we use the definition of an *effective porosity* [4], [20].

For rigid rocks porosity is a static, dimensionless property, but in general ϕ depends on pressure and hence rock should have a parameter referred to as *rock compressibility*

$$c_r = \frac{1}{\phi}\frac{\partial \phi}{\partial p}, \tag{2.1}$$

where p is reservoir pressure [4] and for a rigid rock we obtain $c_r = 0$ and $\phi(p) = const$. If the rock compressibility is constant then porosity can be found as $\phi(p) = \phi_0 e^{c_r(p-p_0)}$.

2.1.2 Permeability

Another important rock parameter is *permeability* $\boldsymbol{K}$. It measures the ability of porous media to conduct flow when pores are already filled with a fluid. The SI unit for permeability is m^2 or 'Darcy' (1 D $\approx 0.987 \cdot 10^{-12}$ m^2) and permeability is rather small value ranging from 10^{-16} m^2 to 20^{-11} m^2 (or from 0.1 mD to 20 D). Permeability is a proportional parameter in Darcy's law (2.6), which will be discussed in more detail in section 2.2, between a flow rate and pressure (or potential gradient $\mathit{\Phi} = \nabla p + \rho g \nabla z$)

$$\vec{v} = -\frac{\boldsymbol{K}}{\mu} \nabla \mathit{\Phi},$$

where μ is the fluid viscosity and $\vec{v}$ is the *superficial velocity* (the flow rate divided by the cross-sectional area).

Temperature and pressure apparently have an affect on fractures and therefore on the permeability, and also the permeability can be found as function of porosity from the Carman-Kozeny relation [6] or from the Fair and Hatch formula [4]. Besides the permeability can be found with the *hydraulic conductivity* κ for a subsurface system which is proportionality constant for the flow of water through a porous media. The permeability can be calculated as follows:

$$\boldsymbol{K} = \kappa \frac{\mu}{\rho g}, \tag{2.2}$$

where ρ is the density of the fluid and g is the acceleration due to the gravity.

2.1.3 Density and viscosity

For an incompressible single-phase flow only the fluid viscosity and the fluid density are needed. If the flow is compressible the fluid compressibility also needs to be considered.

The dynamic *viscosity* (μ) of a fluid measures the fluid resistance to its gradual deformation in SI units [Pa·s] or [Poise]. As an example the liquid such as an olive oil has bigger viscosity than water and therefore moves slower. The viscosity of the fluid is a function of temperature only and in this work we consider the simplest case, when the viscosity is constant.

Another parameter is the *density* of fluid, which equals the mass of the fluid per unit volume ($\rho = \frac{m}{V}$) in the SI-units [kg/m^3]. The density generously can be dependent on temperature and pressure, and conversely the change in density causes fluid pressure and temperature changes $\rho = \rho(p, \mathit{\Theta})$. From the thermodynamics we know such properties as the isothermal compressibility (c_f) and the isobaric thermal expansion coefficient (β) [4]

$$c_f = -\frac{1}{V}\left(\frac{\partial V}{\partial p}\right)_{\mathit{\Theta}}, \qquad \beta = \frac{1}{V}\left(\frac{\partial V}{\partial \mathit{\Theta}}\right)_{p}, \tag{2.3}$$

where V is the volume, p is the pressure, Θ is the temperature and subscripts Θ and p mean the isothermal and isobaric processes, respectively.

The fixed number N of particles implies $\rho V = const$ and thus $d(\rho V) = \rho dV + V d\rho = 0$. In this case the change in volume equals

$$dV = \left(\frac{\partial V}{\partial p}\right)_{\Theta} dp + \left(\frac{\partial V}{\partial \Theta}\right)_{p} d\Theta$$

and then we obtain the formula

$$\frac{d\rho}{\rho} = -\frac{dV}{V} = c_f \cdot dp - \beta \cdot d\Theta. \tag{2.4}$$

Often the density changes very slowly in time and the temperature keeps constant. Hence the term $\beta \cdot d\Theta$ cancels out in the equation (2.4) and we derive formula for the *fluid compressibility*

$$c_f = \frac{1}{\rho}\frac{\partial \rho}{\partial p} = \frac{d\ln(\rho)}{dp}. \tag{2.5}$$

In this formula the relationship between the density ρ and the fluid pressure p is similar to the relationship between the porosity ϕ and the pressure p in the rock compressibility equation (2.1).

2.2 Darcy's Law

The equation

$$\vec{v} = -\frac{\boldsymbol{K}}{\mu}(\nabla p - \rho \vec{g}), \tag{2.6}$$

is known as Darcy's law and was first described by Henry Darcy, a french hydraulic engineer, in 1856 [4]. Darcy made an experiment, in which he designed a vertical tank filled with sand. He injected the water at the tank top that flowed through the whole tank and flowed out at the bottom. After the sand pack was filled with water, Darcy measured the water volumetric flow rate out of the tank. As a final result he noticed the relation between velocity and hydraulic head (pressure):

$$\frac{q}{A} = \kappa \frac{h_t - h_b}{L},$$

where q is the volumetric flow rate in [m^3/s], A is the cross-sectional area, h_t - top hydraulic head, h_b - bottom hydraulic head, L is the length of the tank and

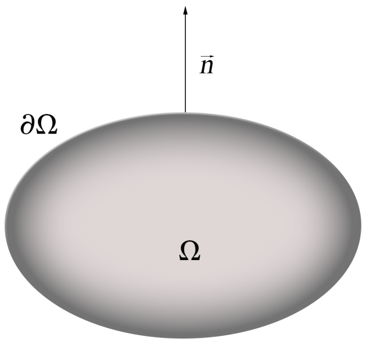


Figure 2.2: Volume Ω with boundary $\partial\Omega$ and outward normal $\vec{n}$.

κ is the hydraulic conductivity. The hydraulic head (h) can be found as energy (E) per mass (m) and gravitational constant (g): $h = E/mg = z + p/\rho g$, here p is the pressure, ρ is the density and z is the depth. The function κ we remember from the intrinsic permeability relation (2.2) and so $\kappa = \rho g \boldsymbol{K}/\mu$.

If we denote the flux as $\vec{v} = \vec{q}/A$, then the equation (2.6) can be derived. Here $\vec{v}$ is the superficial velocity or the *apparent macroscopic* volumetric flux [m/s]. The *macroscopic (intrinsic)* fluid velocity is therefore given by

$$\vec{V} = \vec{v}/\phi. \tag{2.7}$$

The equation (2.6) in such form was derived from the Navier-Stokes equations in 1956 and expresses the conservation of momentum [20]. The Fourier's law for the heat conduction is analogous to Darcy's law and we will use this fact later in solving the energy equation.

2.3 Conservation of Mass

The mass conservation equation states that the accumulation of mass inside some control volume (Figure 2.2) should be equal the net flux over the boundaries of this volume,

$$\int_\Omega \frac{\partial}{\partial t}(\phi\rho)d\Omega = -\int_{\partial\Omega}(\rho\vec{v})\cdot\vec{n}dS + \int_\Omega Q d\Omega, \tag{2.8}$$

where Ω is the domain with the boundary $\partial\Omega$, ϕ is the rock porosity, ρ is the fluid density, $\vec{v}$ is the superficial velocity, $\vec{n}$ is the outward unit normal and Q is the source/sink term in units of [kg/(m^3·s)]. Applying the divergence theorem to the equation (2.8) we obtain the mass conservation equation that holds for any volume Ω

$$\int_\Omega \left(\frac{\partial \phi\rho}{\partial t} + \nabla \cdot (\rho\vec{v}) - Q \right) dV = 0$$

and in the differential form

$$\frac{\partial}{\partial t}(\phi\rho) + \nabla \cdot (\rho\vec{v}) = Q. \tag{2.9}$$

However the system (2.9) contains more unknowns than equations and we need to introduce the constitutive laws to achieve the relationship between different states of the system (pressure, volume, temperature, etc.) and close the mathematical model. Darcy's law is one of such constitutive equations that supplements the mass conservation equation and provides a relation between the superficial velocity $\vec{v}$ and the fluid pressure p.

In general the differential equation (2.9) is a nonlinear hyperbolic equation, because density, viscosity and porosity depend on temperature and/or pressure. By substituting Darcy's law (2.6) into the mass conservation equation (2.9) we obtain

$$\frac{\partial(\phi\rho)}{\partial t} - \nabla \cdot \left(\frac{\rho}{\mu} \boldsymbol{K} (\nabla p - \rho\vec{g}) \right) = Q. \tag{2.10}$$

which is referred to as *the pressure equation*. Assume that the density changes very slowly and so the temperature is a constant and the equation (2.5) works. With the formulas (2.1) and (2.5) we obtain

$$\frac{\partial(\phi\rho)}{\partial t} = \phi\frac{\partial\rho}{\partial p}\frac{\partial p}{\partial t} + \rho\frac{\partial\phi}{\partial p}\frac{\partial p}{\partial t} = c_t\phi\rho\frac{\partial p}{\partial t},$$

where $c_t = c_r + c_f$ denotes the *total compressibility* and the equation (2.10) becomes

$$c_t\phi\rho\frac{\partial p}{\partial t} - \nabla \cdot \left(\frac{\rho}{\mu} \boldsymbol{K} (\nabla p - \rho\vec{g}) \right) = Q. \tag{2.11}$$

If ρ and ϕ are independent of p then rock and fluid compressibilities are $c_f = 0$ and $c_r = 0$. When the density is constant the pressure equation (2.11) becomes an elliptic equation

$$-\rho\nabla \cdot \left(\frac{\boldsymbol{K}}{\mu} (\nabla p - \rho\vec{g}) \right) = Q, \tag{2.12}$$

or

$$\rho\nabla \cdot \vec{v} = Q.$$

The flow with such parameters is referred to as *incompressible flow* and if we have no source/sink term the equation simplifies to $\nabla \cdot \vec{v} = 0$. Both the fluid compressibility and the thermal expansion coefficient are zero when the flow is assumed as incompressible. We will consider the incompressible flow in this thesis.

If $c_f = const$ and independent of pressure, equation (2.5) can be integrated from a known density ρ_0 at a pressure datum p_0 and we get an equation of state for constant compressibility case

$$\rho(p) = \rho_0 e^{c_f(p-p_0)}.$$

When the flow is slightly compressible then the equation of state is $\rho(p) = \rho_0[1 + c_f(p - p_0)]$.

2.3.1 Boundary Conditions

Every natural phenomenon can be described by partial differential equations and every model in porous media characterized by its own equation together with boundary and initial conditions. If we solve a time-dependent problem (2.10) then initial values should be given for each unknown in a model at some initial time $t = t_0$. However the equation (2.12) represents a *boundary-value problem* and we should define only boundary conditions. In reservoir simulation we have Dirichlet and Neumann boundary conditions. A Dirichlet condition describes pressure at the boundaries and that means that there is another aquifer system that gives a pressure support. The form of a Dirichlet condition is

$$p(\vec{x}_D, t) = p_D(\vec{x}_D, t) \text{ for } \vec{x}_D \in \Gamma_D \subset \partial\Omega. \tag{2.13}$$

A Neumann condition (inhomogeneous) describes fluid flow through the boundary, which can be an inflow or an outflow

$$\vec{v} \cdot \vec{n} = u_N(\vec{x}_N, t) \text{ for } \vec{x}_N \in \Gamma_N \subset \partial\Omega. \tag{2.14}$$

The special case for Neumann condition is no-flow boundary condition or homogeneous Neumann condition

$$\vec{v} \cdot \vec{n} = 0 \text{ for } \vec{x}_N \in \Gamma_N \subset \partial\Omega. \tag{2.15}$$

However if we have only Neumann boundary conditions in the model, the solution of equation (2.12) is defined only up to a constant, unless we prescribe a datum value at some internal point or along the boundary. This situation violates the uniqueness requirement from the definition of a well-posed problem [4]:

1. The solution should exist (*existence*).
2. The solution should be uniquely determined (*uniqueness*).
3. The solution should depend continuously on the data (*stability*).

2.3.2 Well model

A well source is a more complicated model then simple source initialization regarding the implementation [26], [15], [5]. Physically a well is a pipe in the ground that allows a fluid to be injected to the subsurface and conversely a well allows a fluid from the subsurface to be extracted. If a well is drilled vertically then a fluid flows radially in the formation. In many simulations the mathematical representation of wells determines the well as a separate structure and not a part of the formation. To describe the relation between the bottom-hole pressure of a well and the flow rate the *inflow performance relation* (IPR) is used. We can assume that a volumetric pumping rate is proportional to the difference between the average reservoir pressure p_R in the grid cell and the bottom-hole pressure p_{bh} in the well [2]

$$q = J(p_R - p_{bh}). \tag{2.16}$$

Here J is the constant of proportionality and in the MATLAB Reservoir Simulation Toolbox, which we use for our simulations, it is referred to as a *well injectivity index* [21],[20]. To calculate J we need to describe a radial flow with Darcy's law.

Assume that a fluid moves radially toward the wellbore and cylindrical flow in the reservoir where flow is horizontal across the formation

$$v = \frac{qB}{2\pi r h},$$

where B is the formation volume factor, $B = 1$ for an incompressible flow, h is the height and r is the well radius.

Consider a vertical well in the porous medium with a uniform permeability $\boldsymbol{K}$. From Darcy's law we have

$$v = \frac{\boldsymbol{K}}{\mu}\frac{dp}{dr}.$$

Thus we can get an equation

$$\frac{2\pi \boldsymbol{K} h}{q\mu B}\frac{dp}{dr} = \frac{1}{r}$$

and integrate it from the *wellbore radius* r_w to the *drainage boundary* r_e, which can be found as $r_e \approx 0.2\sqrt{\Delta x \Delta y}$

$$p_e = p_{bh} + \frac{q\mu B}{2\pi \boldsymbol{K} h} \ln(r_e/r_w).$$

After a few transformations [20] and adding the skin factor S we obtain formula for the well injectivity

$$J = \frac{2\pi \boldsymbol{K} h}{\mu B(\ln(r_e/r_w) - 0.75 + S)}. \tag{2.17}$$

2.4 Time of Flight

Often in reservoir modelling the visualization of a flow field is needed and vector field lines are commonly used for these purposes. There are three types of field lines: *streamlines*, *streaklines* and *pathlines*, which are identical for a steady flow [4].

Streamlines are far more used in the reservoir simulation than streaklines or pathlines. Streamlines is like an instantaneous snapshot of the flow field and the result is a bunch of curves that are tangential to the velocity $\vec{v}$. The streamline formula is

$$\frac{d\vec{x}}{dr} \times \vec{v}(\vec{x}, t) = 0, \text{ or } \frac{d\vec{x}}{dr} = \frac{d\vec{v}(t)}{|\vec{v}(t)|} \tag{2.18}$$

where $\vec{x}(r)$ is a parametric representation of a single streamline at the time t.

There is another way to parametrize streamlines by using time-of-flight τ. *Time-of-flight* is a parameter which is used to express the time it takes a fluid particle to cover a distance r along streamline. It can be found from this differential equation [10], [17]

$$\vec{v} \cdot \nabla \tau = \phi, \tag{2.19}$$

or equivalently from the parametric equation

$$\tau(r) = \int_0^r \frac{\phi(\vec{x}(s))}{|\vec{v}(\vec{x}(s))|} ds. \tag{2.20}$$

Chapter 3

Heat transfer

In this chapter we will discuss the main principles of the heat transfer in porous media. There are three different types of heat transfer that are usually referred to as *modes of heat transfer*:

1) *Conduction*, which occurs at the molecular level. The main condition for the conduction is temperature gradient in a medium and heat is transferred along that gradient.
2) *Convection* happens by random molecular motion (diffusion) or by the bulk motion of a fluid, that carries energy from one place to another (advection).
3) *Radiation* is a heat transfer, when a heat energy is transferred by electromagnetic waves through medium or space between surfaces. In this thesis we neglect the effect of the radiation heat flow in geothermal reservoirs.

There are four laws of thermodynamics that we should know, when we talk about the heat transfer. *The zeroth law* claims that if two systems are in the thermal equilibrium with third system, then all three systems are in thermal equilibrium with each other. *The first law of thermodynamics* states that the sum of heat amount supplied to an isolated system and the amount of mechanical work done by the system equals to the change in the internal energy of the system. Mathematically the first law can be written as $dU = \delta Q - \delta W$, where δQ is the amount of energy added to system by a heating, dU is the change of the internal energy and δW is the amount of energy which was lost by the system due to work [22]. The second and third laws of thermodynamics are related to the *entropy* of an isolated system which was introduced as a measure of change in thermal energy ΔQ of the system per temperature ($\Delta S = \Delta Q/\Theta$). *The second law of thermodynamics* claims that the entropy of an isolated system can not decrease and from *the third law* entropy tends to a constant value while the temperature of the system approaches the absolute zero.

3.1 Conduction

Conduction or *diffusion* is the heat transfer which occurs between particles with different levels of energy within a body. The conductive transfer often appears in solid materials due to vibrations of particles and free electrons, and even in fluid collisions of molecules take place. Conduction within fluids is also very significant and this is the way heat reaches and leaves the surface of a solid.

Table 3.1: Thermal conductivities in W/(m · K) [36]

Material	**Value**
Granite	1.7 – 4.0
Limestone	1.26 – 1.33
Rock, solid	2 – 7
Rock, porous volcanic	0.5 – 2.5
Sand, dry	0.15 – 0.25
Sand, saturated	2 – 4
Sandstone	1.7
Soil, saturated	0.6 – 4
Water	0.58

Mathematically the conductive heat transfer is described by Fourier's law and expresses that the heat flux is proportional to the temperature gradient

$$\vec{\psi} = -\boldsymbol{k}\nabla\Theta, \tag{3.1}$$

where $\vec{\psi}$ is the conductive heat flux or the thermal power per unit area $\vec{\psi} = \vec{\Psi}/A$ in the SI unite [W/m^2], $\boldsymbol{k}$ is the thermal conductivity and Θ is the temperature. The negative sign here points out that the heat flow goes in the contrary direction of a temperature gradient. Equation (3.1) is known as *Fourier's first law of heat conduction.*

The *thermal conductivity* [W/(m · K)] is a second order tensor (like permeability in Darcy's law), it is a property of material and usually depends on pressure and temperature. The thermal conductivity values of some typical materials are presented in Table 3.1. To simplify the problem we assume that the thermal conductivity is constant and does not depend on any parameters.

3.2 Convection

Convection is the heat transport occurred by the molecular motion of fluids or gases and the bulk fluid motion and because of particle movement it is only the property of fluids and gases. Consequently convection involves *advection* as fluid motion and *diffusion* as heat exchange between a surface and a fluid. There are two types of convection: natural due to buoyancy forces (warming up or cooling) and forced due to any external source or mechanism like a fan or a pump).

The heat flux [W/m^2] due to the *advection* can be found as

$$\vec{\psi} = e\vec{V}, \tag{3.2}$$

where e is the thermal energy density in SI units [J/m^3], $\vec{V}$ is the intrinsic velocity (2.7) in [m/s]. The alternative way is to express the heat flux in terms of specific heat capacity.

Table 3.2: Specific heat capacities in J/(kg · K) [34], [32]

Material	**Value**
Granite	790
Limestone	908
Sand	830
Sandstone	920
Soil,dry	800
Soil, wet	1480
Water	4182

The *heat capacity* [J/K] is a physical quantity equal to the amount of heat (δQ) enough to achieve the change in temperature ($d\Theta$) of the system

$$C = \frac{\delta Q}{d\Theta}.$$

The *specific heat capacity* is the heat capacity per a unit of mass

$$c = \frac{\delta Q}{m \cdot d\Theta}, \tag{3.3}$$

and in thermodynamics accepted the property relations that define the specific heat capacity at constant volume $c_V = \left(\frac{\partial u}{\partial \Theta}\right)_V$ and the specific heat capacity at the constant pressure $c_p = \left(\frac{\partial h}{\partial T}\right)_p$. The typical specific heat capacity values are shown in Table 3.2.

Assume the pressure vary slowly in time and in terms of thermal energy density we obtain from (3.3) $c_p = de/(\rho \cdot d\Theta)$ and thus the heat flux equals

$$\vec{\psi} = \rho c_p \vec{v} \Theta, \tag{3.4}$$

where $\vec{\psi}$ is advective heat transfer [W/m^2], ρ is the density, c_p is the heat capacity, $\vec{v}$ is the fluid velocity and Θ is the temperature.

The heat flux [W/m^2] due to diffusion can be found as

$$\vec{\psi} = h \cdot \delta\Theta = h \cdot (\Theta_s - \Theta_f), \tag{3.5}$$

where $\delta\Theta$ is a temperature difference between surface and fluid, and h is the convective heat transfer coefficient in SI units [W/(m$^2\cdot$ K)] [37]. The equation (3.5) is referred to as the *Newton rate equation*. However this law is only valid for some idealized cases and in practice the local thermal equilibrium can be assumed.

3.3 The conservation of energy

The conservation of energy law or *the first law of thermodynamics* states that the total energy of an isolated system is constant (Figure 3.1). The total energy of the system can be divided into internal energy end mechanical energy which, in turn can be divided into potential and kinetic energy. Due to the small velocities in the porous media flow we can neglect kinetic energy and if we do not include gravity forces into the system then we can neglect the potential energy. The internal energy also consists of different types of energy, however we assume that there is only thermal energy. And from the first law of thermodynamics we can conclude that change in thermal energy equals the sum of energy transport (advection),

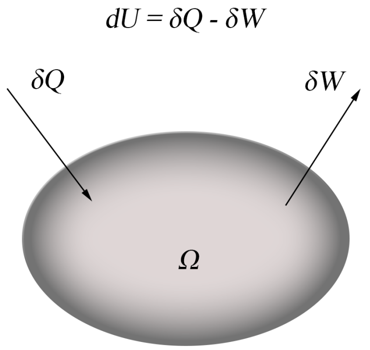


Figure 3.1: The conservation of energy law.

conductive heat flux and heat flux through sources and boundaries. The equation for the fluid phase will be read as

$$\phi\frac{d}{dt}\int_{\Omega}(e)_f dx = -\underbrace{\phi\int_{\partial\Omega}\vec{\psi}_a\cdot\vec{n}ds}_{\text{advective term}} - \underbrace{\phi\int_{\partial\Omega}\vec{\psi}_c\cdot\vec{n}ds}_{\text{conductive term}} + \int_{\Omega}\varPsi_f dx,$$

where ϕ is the porosity, e is the energy density, subscript f means fluid phase, $\vec{\psi}_a$ is the advective heat flux, $\vec{\psi}_c$ is the conductive heat flux and $\vec{\varPsi}_f$ is the source term

$$\vec{\varPsi}_f = \vec{\varPsi}_{cond} + \phi\vec{\varPsi}_{source} = Ah(\Theta_s - \Theta_f) + \phi\varPsi_{source}.$$

However often local thermal equilibrium (LTE) is assumed in studies of heat transfer. It means that in the REV the solid temperature is equal to the fluid temperature ($\Theta_s = \Theta_f$), and if the temperature difference in REV is smaller then temperature difference in the system then the system approaches the local thermal equilibrium. In terms of LTE $\vec{\varPsi}_f = \phi\vec{\varPsi}_{source}$ and from equations (3.1) and (3.2) we get

$$\phi\frac{d(e)_f}{dt} + \phi\nabla\cdot((e)_f\vec{V}) - \phi\nabla\cdot(\boldsymbol{k}_f\nabla\varTheta_f) = \varPsi_f.$$

The equation for the solid phase looks the same as the equation for the fluid except that it does not include the advective term and multiplied on $(1-\phi)$

$$(1-\phi)\frac{d(e)_s}{dt} - (1-\phi)\nabla\cdot(\boldsymbol{k}_s\nabla\varTheta_s) = \vec{\varPsi}_s,$$

where the subscript s means the solid phase and $\varPsi_s = (1-\phi)\varPsi_{sourse}$. After we sum the equations for the solid and fluid phases and obtain the energy balance equation

$$\frac{\partial}{\partial t}\int_{\Omega}((1-\phi)(e)_s+\phi(e)_f)\,d\Omega+\int_{\partial\Omega}((e)_f\vec{v}-(\boldsymbol{k}_{eff}\nabla\Theta)\cdot\vec{n}dS=\Psi_{eff}, \qquad (3.6)$$

where $\boldsymbol{k}_{eff}$ is the effective thermal conductivity and Ψ_{eff} is the effective source term:

$$\boldsymbol{k}_{eff}=(1-\phi)\boldsymbol{k}_s+\phi\boldsymbol{k}_f,$$
$$\vec{\Psi}_{eff}=\vec{\Psi}_s+\Psi_f.$$

If we use the specific heat capacity definition (3.3) and equation (3.4) and apply the divergence theorem, then we will get an expression for the energy equation in differential form [25]

$$(\rho c_p)_{eff}\frac{\partial\Theta}{\partial t}+(\rho c_p)_f\vec{v}\cdot\nabla\Theta-\nabla\cdot(\boldsymbol{k}_{eff}\nabla\Theta)=\Psi_{eff}, \qquad (3.7)$$

where

$$(\rho c_p)_{eff}=(1-\phi)(\rho c_p)_s+\phi(\rho c_p)_f.$$

3.3.1 Initial and boundary conditions

For the energy conservation equation we should initialize boundary and initial conditions as we did in Chapter 2. Let us take the reservoir temperature at some initial time $t=t_0$ as initial value

$$\Theta(\vec{x},t_0)=\Theta_0(\vec{x}) \text{ for } \vec{x}\in\Omega.$$

In this thesis we use two types of boundary conditions: Dirichlet and Neumann. The form of the first-type boundary condition is

$$\Theta(\vec{x}_D,t)=\Theta_D(\vec{x}_D,t) \text{ for } \vec{x}_D\in\Gamma_D\subset\partial\Omega.$$

The Neumann conditions describe the heat flux through the boundary. In general case they are inhomogeneous

$$\vec{\psi}\cdot\vec{n}=\psi_N(\vec{x}_N,t) \text{ for } \vec{x}_N\in\Gamma_N\subset\partial\Omega,$$

however the flux boundary conditions can equal zero

$$\vec{\psi}\cdot\vec{n}=0 \text{ for } \vec{x}_N\in\Gamma_N\subset\partial\Omega.$$

3.4 Dimensionless numbers

The **Reynolds number** is the ratio between inertial forces and viscous forces and defined as

$$Re = \frac{\rho \vec{v} L}{\mu}$$

where L is the characteristic length, ρ is the density $\vec{v}$ is the velocity and μ is the viscosity.

The **Prandtl number** is the ratio of momentum diffusivity to thermal diffusivity and equals to

$$Pr = \frac{c_p \mu}{\boldsymbol{k}},$$

where c_p is the heat capacity and $\boldsymbol{k}$ is the thermal conductivity.

The **Péclet number** is defined as

$$Pe = \frac{L \vec{v} \rho c_p}{\boldsymbol{k}} = Re \cdot Pr. \tag{3.8}$$

Thus the Péclet number shows which term (conductive or convective) dominates in the conservation of energy equation (3.7). We will return to the Péclet number in Chapter 6, where we will run simulations for various thermal conductivities and see how accuracy of the solutions on coarse grids depends on the dominance term in the conductive-convective energy equation.

Chapter 4

Numerical Methods

Partial differential equations that were presented in the former chapters usually require numerical solutions because analytical solutions can be found only for some simplified cases [11]. Different equations have different characteristics and we must use various discretization techniques for parabolic, elliptic or hyperbolic equations. Spatial and temporal discretizations are the main conditions that have influence on the computational time and computational error. Due to the fact that computer resources are limited we need to choose a reasonable resolution to balance accuracy and computational cost, and we should be aware of numerical errors that appear when the grid size or the time step are too large. For instance the pressure equation (2.10) which is a parabolic equation is usually solved implicitly and, by contrast, a transport equation should be discretized explicitly to reduce the numerical dispersion. To solve the energy equation (3.7) we can treat the temperature both explicitly or implicitly depending on domination of convective or conductive term [29].

4.1 Discretization in space

Finite volume methods (FVM) are widely used in solving porous media problems along with finite difference and finite elements methods due to its ability to keep local properties of conservation laws. We consider that Ω is the whole domain (reservoir domain) and it is represented by a grid structure and divided on small grid cells Ω_i referred to as *control volumes.*

To apply finite volume scheme the hyperbolic-parabolic equation (3.7) should be integrated over each control volume Ω_i. For every divergence term we can apply the divergence theorem and obtain a surface integral which is representing flux over the surface $\partial\Omega_i$, $i = 1, \ldots, n$

$$\int_{\Omega_i} \frac{\partial \Theta}{\partial t} d\Omega + \int_{\partial\Omega_i} \frac{(\rho c_p)_f \Theta \vec{v}}{(\rho c_p)_{eff}} \cdot \vec{n} dS - \int_{\partial\Omega_i} \frac{\boldsymbol{k}_{eff} \nabla \Theta}{(\rho c_p)_{eff}} \cdot \vec{n} dS = \int_{\Omega_i} \frac{\Psi}{(\rho c_p)_{eff}} d\Omega. \quad (4.1)$$

The obtained system of equations (4.1) depends on a finite set of unknowns and we suppose that PDE (3.7) is solved over Ω if we have solutions over each control volume Ω_i. In this thesis we consider that the system of equations (4.1) is linear and thus we can use one of the linear solvers to compute the temperature. However in many cases we deal with the nonlinear system of equations and iterative schemes like Newton-Raphson method can be used to solve such systems.

4.1.1 Two-Point Flux Approximation

To solve the pressure equation (2.12) or to discretize the conductive term in the energy conservation equation (3.7) we are using TPFA method, which is widely used due to its simplicity, computational efficiency and robustness. Assume constant viscosity and density and no gravity forces in the system (2.12), hence we obtain the equation

$$-\nabla \cdot (\boldsymbol{K} \nabla p) = Q. \quad (4.2)$$

Face velocities in the equation (4.2) should sum to zero in the whole domain, this requirement means the fulfilment of the mass conservation law and we will get a physical solutions in this case.

At first, we integrate equation (4.2) over every single cell Ω_i, then we can apply the divergence theorem and Darcy's law (2.6)

$$\int_{\partial\Omega_i} (\vec{v} \cdot \vec{n}) dS = \int_{\Omega_i} Q d\Omega, \quad (4.3)$$

where $\vec{v} = \boldsymbol{K}\nabla p$ is the velocity and $\vec{n}$ is the outward unit normal vector. The obtained equation (4.3) is known as the conservation law and here the flux over each side can be approximated by the neighbouring pressures:

$$\int_{\Gamma_{ik}} (\vec{v} \cdot \vec{n}) dS \approx \sum_{j=1}^{\nu} T_j p_j, \qquad \Gamma_{ik} = \partial\Omega_i \cap \partial\Omega_k. \quad (4.4)$$

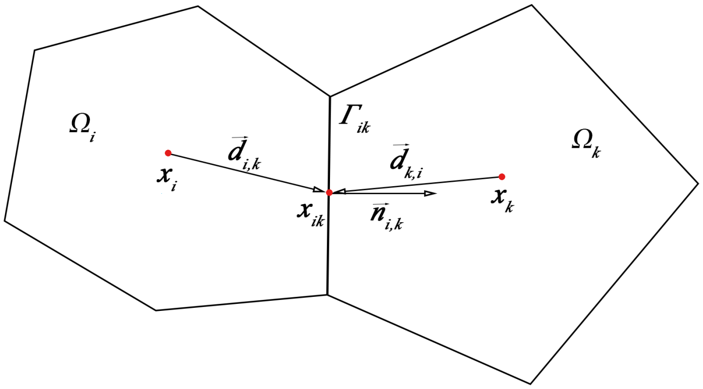


Figure 4.1: Two-point flux discretization for grid cells Ω_i and Ω_k.

It is simply stated: we need to solve the linear system $T \cdot p = q$ to find the pressure, where T is $n \times n$ matrix, p and q are $n \times 1$ vectors.

Let us introduce the unknown pressure $\hat{p}_{i,k}$ at the shared edge center, we can approximate the pressure gradient by the difference between the pressure at the cell center and the pressure at the face center. The flux from cell Ω_i to cell Ω_k over the face with half transmissibilities $\alpha_{i,k}$ equals

$$f_{i,k} = -\int_{\Gamma_{ik}} \vec{n}_{i,k} \cdot (\boldsymbol{K}_i \nabla p) dS \approx \alpha_{i,k}(p_i - \hat{p}_{i,k}), \tag{4.5}$$

$$\alpha_{i,k} = \frac{A_{i,k} \vec{n}_{i,k} \cdot \boldsymbol{K}_i}{\vec{d}_{i,k} \cdot \vec{d}_{i,k}} \cdot \vec{d}_{i,k}. \tag{4.6}$$

$A_{i,k}$ is the area of shared edge, $\boldsymbol{K}_i$ is the permeability in cell Ω_i, $\vec{d}_{i,k}$ is the distance vector from centroid of cell i to the face center, $\hat{p}_{i,k}$ is the pressure at the face center Γ_{ik} and p_i is the pressure at the center of the cell Ω_i (Figure 4.1).

The flux from cell Ω_K to cell Ω_i can be found similarly as

$$f_{k,i} = -\int_{\Gamma_{ik}} \vec{n}_{k,i} \cdot (\boldsymbol{K}_k \nabla p) dS \approx \alpha_{k,i}(p_k - \hat{p}_{k,i}). \tag{4.7}$$

Keeping in mind that $f_{i,k} = -f_{k,i}$ we denote $f_{ik} := f_{i,k}$ and obtain

$$\alpha_{i,k}^{-1} f_{ik} = \alpha_{i,k}^{-1} f_{i,k} = p_i - \hat{p}_{i,k}, \qquad -\alpha_{k,i}^{-1} f_{ik} = \alpha_{k,i}^{-1} f_{k,i} = p_k - \hat{p}_{k,i}.$$

Considering that the pressure at common face Γ_{ik} is identical $\hat{p}_{k,i} = \hat{p}_{i,k}$, we get formula for the flux across interface Γ_{ik} between the cells Ω_i and Ω_k

$$f_{ik} = T_{ik}(p_i - p_k), \qquad T_{ik} = [\alpha_{i,k}^{-1} + \alpha_{k,i}^{-1}]^{-1}, \tag{4.8}$$

where T_{ik} is the face transmissibility. Inserting flux from the equation (4.8) into (4.2) we will get a system of equations which can be easily solved

$$\sum_{k=1}^{n_i} T_{ik}(p_i - p_k) = Q_i, \qquad \forall \Omega_i \subset \Omega, i = 1, \ldots, n, \tag{4.9}$$

where n_i is the face number for the cell Ω_i and n is the total number of grid cells.

Analogous to the calculation made for pressure equation, we obtain similar TPFA discretization scheme for temperature in the heat equation conductive term

$$\sum_{k=1}^{n_i} T_{ik}(\Theta_i - \Theta_k) = \Psi_i, \qquad \forall \Omega_i \subset \Omega, \tag{4.10}$$

but here we use the effective thermal conductivity $\boldsymbol{k}^{eff}$ instead of permeability tensor to compute the face transmissibilties:

$$T_{ik} = [\alpha_{i,k}^{-1} + \alpha_{k,i}^{-1}]^{-1}, \qquad \alpha_{i,k} = \frac{A_{i,k}\vec{n} \cdot \boldsymbol{k}_i^{eff}}{\vec{d}_i \cdot \vec{d}_i} \cdot \vec{d}_i. \tag{4.11}$$

4.1.2 Upwind scheme

The upwind method we use to discretize the advective term of the energy conservation equation. This method is appropriate because it recognizes the flow direction as the inflow or outflow through the edge and based on this conclusion adds the positive or negative sign to the variable. Consider the divergence

$$\nabla \cdot (\Theta \vec{v}) = \Psi. \tag{4.12}$$

Let us integrate it over every grid cell Ω_i and apply the divergence theorem [11]

$$\int_{\partial \Omega_i} (\Theta \vec{v}) \cdot \vec{n} dS = \int_{\Omega_i} \Psi d\Omega_i, \qquad \forall \Omega_i \subset \Omega. \tag{4.13}$$

We shall consider in detail the flux through internal edges. Assume Γ_{ik} is the common face of the two grid cells Ω_i and Ω_k. The flux over the edge Γ_{ik} in the direction of the normal $\vec{n_{ik}}$ equals $\vec{v_{ik}} \cdot \vec{n_{ik}}$ and thus the heat flux through the edge Γ_{ik} will be equal

$$\int_{\partial\Omega_i} \mathit{\Theta} \vec{v} \cdot \vec{n} dS \approx \begin{cases} \mathit{\Theta}_i \int_{\Gamma_{ik}} \vec{v}_{ik} \cdot \vec{n}_{ik} dS, & \text{if } \vec{v} \geq 0, \\ -\mathit{\Theta}_k \int_{\Gamma_{ik}} \vec{v}_{ik} \cdot \vec{n}_{ik} dS, & \text{if } \vec{v} < 0. \end{cases} \tag{4.14}$$

4.2 Discretization in Time

The pressure equation (2.12) is a boundary-value problem and only needs to be solved once. The heat equation (3.7) on the contrary needs temporal discretization. We can write down the equation (3.7) in simple form

$$\frac{\partial}{\partial t} \int_{\Omega_i} \mathit{\Theta} \mathrm{d}\Omega = \int_{\Omega_i} F(\mathit{\Theta}) \mathrm{d}\Omega,$$

and for rather small control volume we can approximately compute the integral as volume of the cell V_i multiplied by the integration element and thus we obtain an autonomous system of differential equations

$$\frac{\mathrm{d}\mathit{\Theta}_i}{\mathrm{d}t} \approx F_i(\Theta), \tag{4.15}$$

where F does not depend explicitly on t and such system can be solved with various types of numerical methods [13]. However in this thesis we are going to solve the equation with one-step schemes: *Euler's Forward method* and *Euler's Backward method*.

To describe one-step method on the example of equation (4.15) assume a generic point $t \in [t_0, t_f]$ and define a single step of the method

$$\mathit{\Theta}_{next} = \mathit{\Theta} + \Delta t \mathit{\Phi}(t, \mathit{\Theta}; \Delta t), \qquad \mathit{\Theta} \in \mathbb{R}^d$$

where $\mathit{\Phi}$ if the function that defines the method. The Euler's Forward method is one of the oldest, which was proposed by Euler in 1768. The general formula for our equation will be

$$\mathit{\Theta}^{n+1} = \mathit{\Theta}^n + \Delta t F(t_n, \mathit{\Theta}^n), \tag{4.16}$$

the method function $\mathit{\Phi}$ does not depend on Δt. The value $\mathit{\Theta}^n$ is an approximation of the analytical solution for the ODE system at the time step t_n ($\mathit{\Theta}^n \approx \mathit{\Theta}(t_n)$).

Consider the implicit r-stage Runge-Kutta method, which is more general case of an explicit r-stage Runge-Kutta method

$$\varPhi(t, \varTheta; \Delta t) = \sum_{s=1}^{r} \alpha_s \kappa_s(t, \varTheta; \Delta t),$$

$$\kappa_s = F\left(t + \mu_s \Delta t, \varTheta + \Delta t \sum_{j=1}^{r} \lambda_{sj}\kappa_j\right), \ s = 2, 3, .., r.$$

with the conditions $\mu_s = \sum_{j=1}^{s-1} \lambda_{sj}$, $\sum_{s=1}^{r} \alpha_s$. The Euler's Backward scheme is the simplest example of an implicit Runge-Kutta method:

$$\varTheta^{n+1} = \varTheta^n + \Delta t \kappa_1,$$

$$\kappa_1 = F(t_n + \Delta t, \varTheta^n + \Delta t \kappa_1),$$

or

$$\varTheta^{n+1} = \varTheta^n + \Delta t F(t_n + \Delta t, \varTheta^{n+1}). \tag{4.17}$$

Generally the heat equation (3.7) is nonlinear, the heat parameters such as the thermal conductivity or heat capacity usually depend on temperature and thus fix-point iteration method or Newton-Raphson scheme is needed to approximate $\varTheta^{n+1}$. In cases where the nonlinearity is strong, the scheme may not converge at all and a reduced time-step is used. In this thesis we assume that such heat parameters equal constant and function F is linear in the heat equation.

LU solver is effective method to find the solution of the linear system in Matlab. If we could compute the inverse matrix A^{-1} then it will be easy to find the solution $\varTheta = A^{-1}b$. However the matrix decomposition into the lower and upper triangular matrices is more efficient way to obtain the solution [19]

$$\begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{pmatrix} = \begin{pmatrix} 1 & 0 & \dots & 0 \\ l_{21} & 1 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ l_{n1} & l_{n2} & \dots & 1 \end{pmatrix} \cdot \begin{pmatrix} u_{11} & u_{12} & \dots & u_{1n} \\ 0 & u_{22} & \dots & u_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & u_{nn} \end{pmatrix}$$

Thus when matrix $A = LU$, the equation $A \cdot \vec{x} = \vec{b}$ can be solved as

$$\begin{cases} L \cdot \vec{y} = & \vec{b}, \\ U \cdot \vec{x} = & \vec{y}. \end{cases}$$

4.3 Implementation of Boundary and Initial conditions

In this section we will describe the implementation of boundary conditions in equation (2.12) and implementation of boundary and initial conditions in discretized equation (3.7). In MRST boundary conditions are defined with the two types 'pressure' and 'flux' and correspond to Dirichlet and Neumann boundary conditions respectively. For every boundary face we set a value which is measured in the units of [Pa] for pressure and [m^3/s] for flux. In addition, boundary condition per grid face should be defined uniquely and in case of problem with only Neumann conditions the boundary fluxes should sum to zero for incompressible flow. If we do not set up boundary condition at some outer edge in MRST then this edge is defined as no-flow boundary.

Let us set the problem for the pressure equation (2.12) in domain Ω. Consider $\Gamma = \partial\Omega$ is the whole boundary of the domain Ω, that can be divided into the Dirichlet and Neumann boundaries $\Gamma = \Gamma_D \cap \Gamma_N$. Dirichlet condition is given in the form of $p = p_D$ in Γ_D (2.13) and Neumann in the form of $\vec{v} \cdot \vec{n} = f_N$ in Γ_N (2.14). For every single grid cell $\Omega_i \subset \Omega$ the sum of the flux over the edges should be equal to the source term in this cell. Let us denote ω as a set of all edges in Ω ($\omega = \omega_{internal} \cup \omega_{external}$) and ω_i as a set of all edges in the cell Ω_i and then to find pressures we need to solve the system

$$\sum_{\gamma \in \omega_i} f_{i,\gamma} = Q_i, \qquad \forall \Omega_i \subset \Omega, \tag{4.18}$$

where $f_{i,\gamma}$ is the flux across the edge γ in the cell Ω_i,

$$f_{i,\gamma} = \begin{cases} (p_i - p_j) \cdot T_{ik}, & \text{for } \gamma \in \omega_i \cap \omega_j \cap \omega_{\text{internal}}, \\ (p_i - \hat{p}_{i,D}) \cdot \alpha_{i,D}, & \text{for } \gamma \in \omega_i \cap \Gamma_D, \\ \displaystyle\int_\gamma f_{i,N} dS, & \text{for } \gamma \in \omega_i \cap \Gamma_N. \end{cases}$$

Here T_{ik} is the face transmissibility on the internal edge between the two cells Ω_i and Ω_k, $\hat{p}_{i,D}$ is the pressure at the external face center $\boldsymbol{x}_\gamma$, $\alpha_{i,D}$ is the half transmissibility on the Dirichlet boundary

$$\alpha_{i,D} = \frac{A_{i,\gamma} \boldsymbol{K}_i \vec{n}_{i,\gamma} \cdot \vec{d}_{i,\gamma}}{\|\vec{d}_{i,\gamma}\|^2},$$

A_γ is the area of the edge γ and $f_{i,N}$ is the flux across the Neumann boundary. The source term Q in MRST can be defined as a volumetric flow rate in SI units [m^3/s]. However it is important to know that there can be only one net source per grid cell and the source terms must sum to zero for the incompressible flow with no-flow boundary conditions. If this condition is not observed we will get an ill-posed model with the inactive mass conservation law.

When we start to solve the temperature equation, we should discretize conductive and convective parts separately. At first we should define the initial condition. For every grid cell Ω_i we set temperature value Θ_i^0 at initial time t_0.

$$\Theta_i(t_0) = \Theta_i^0, \qquad \forall \Omega_i \subset \Omega. \tag{4.19}$$

We discretize the conservation of energy equation (4.1) for each grid cell as follows

$$\begin{aligned} \frac{\Theta_i^{n+1} - \Theta_i^n}{dt} V_i + \frac{(\rho c_p)_f}{(\rho c_p)_{eff}} \sum_{\gamma \in \omega_i} \psi_{i,\gamma}^{adv} + \frac{1}{(\rho c_p)_{eff}} \sum_{\gamma \in \omega_i} \psi_{i,\gamma}^{cond} = \\ \frac{(\rho c_p)_f}{(\rho c_p)_{eff}} \Phi_i V_i + \frac{1}{(\rho c_p)_{eff}} \Psi_i V_i, \qquad \forall \Omega_i \subset \Omega, \end{aligned} \tag{4.20}$$

where V_i is the volume of cell Ω_i. The parameter $\Phi = \Theta \cdot Q$ was obtained from the advective term, which is represented as product of velocity and temperature gradient. To apply the divergence theorem we need to transform the product

$$\vec{v} \cdot \nabla \Theta = \nabla \cdot (\vec{v}\Theta) - \Theta \nabla \cdot \vec{v} = \nabla \cdot (\vec{v}\Theta) - \Theta Q,$$

where Q is the source/sink term from the equation (2.12) per density and thus we obtain the equation

$$\frac{(\rho c_p)_f}{(\rho c_p)_{eff}} \int_{\partial \Omega_i} \Theta \vec{v} \cdot \vec{n} dS = \frac{(\rho c_p)_f}{(\rho c_p)_{eff}} \int_{\Omega_i} \Phi d\Omega.$$

The implementation of the conductive term is similar to the pressure equation, consider

$$-\frac{1}{(\rho c_p)_{eff}} \int_{\partial \Omega_i} \boldsymbol{k}_{eff} \nabla \Theta \cdot \vec{n} dS.$$

If we denote $\psi_{i,\gamma}^{cond}$ as the heat flux through the edge γ, T_{ik} as the face transmissibility for conductivity on the internal edge, $\alpha_{i,D}$ as the half transmissibility on Dirichlet boundary

$$\alpha_{i,D} = \frac{A_{i,\gamma} \boldsymbol{k}_i^{eff} \vec{n}_{i,\gamma} \cdot \vec{d}_{i,\gamma}}{\|\vec{d}_{i,\gamma}\|^2},$$

Θ_i as the cell temperature and $\Theta_{i,D}$ as the temperature on the Diriclet boundary, then we can represent the conductive term as the sum of the conductive fluxes $\psi_{i,\gamma}^{cond}$ over every edge in the cell Ω_i, where

$$\psi_{i,\gamma}^{cond} = \begin{cases} (\Theta_i - \Theta_j) \cdot T_{ik}, & \text{for } \gamma \in \omega_i \cap \omega_j \cap \omega_{\text{internal}}, \\ (\Theta_i - \Theta_{i,D}) \cdot \alpha_{i,D}, & \text{for } \gamma \in \omega_i \cap \Gamma_D, \\ \displaystyle\int_\gamma \psi_{i,N} dS, & \text{for } \gamma \in \omega_i \cap \Gamma_N. \end{cases}$$

The system for the advective part in equation (3.7) differs from the conductive part. We can represent the term

$$\frac{(\rho c_p)_f}{(\rho c_p)_{eff}} \int_{\Omega_i} \vec{v} \cdot \nabla \Theta dS$$

as the sum of advective heat fluxes $\psi_{i,\gamma}^{adv}$ over every edge in the cell Ω_i, where

$$\psi_{i,\gamma}^{adv} = \begin{cases} \Theta_i \displaystyle\int_\gamma \vec{v}_{i,\gamma} \cdot \vec{n}_{i,\gamma} dS, & \text{if } \vec{v} \geq 0, \\ -\Theta_i^* \displaystyle\int_\gamma \vec{v}_{i,\gamma} \cdot \vec{n}_{i,\gamma} dS, & \text{if } \vec{v} < 0. \end{cases}$$

We choose the parameter Θ_i^* according to the type of a face. In case of an internal face we choose Θ_i^* as the neighbouring cell temperature, for the first-type boundary face it equals to the boundary temperature and for Neumann boundary we compute it as follows

$$\Theta_i^* = \Theta_{i,N} - \psi_{i,N} \frac{\|\vec{d}_{i,\gamma}\|^2}{A_{i,\gamma} \boldsymbol{k}_i^{eff} \vec{n}_{i,\gamma} \cdot \vec{d}_{i,\gamma}}.$$

Hence the discretization scheme for Euler's Forward method is

$$\Theta_i^{n+1} = \Theta_i^n - \frac{dt}{V_i}(A_1 + A_2) \cdot \Theta_i^n + \frac{dt}{V_i} B_i, \quad \forall \Omega_i \subset \Omega, \tag{4.21}$$

and for Euler's Backward Scheme

$$\Theta_i^{n+1} + \frac{dt}{V_i}(A_1 + A_2) \cdot \Theta_i^{n+1} = \Theta_i^n + \frac{dt}{V_i} B_i, \quad \forall \Omega_i \subset \Omega, \tag{4.22}$$

where A_1 and A_2 are the matrices for the advective and conductive terms and B is the formula's (4.20) right hand side.

4.4 Convergence

Consider the explicit and implicit schemes for the 1D problem

$$u_i^{n+1} - u_i^n = \Delta t F(u^n, \Delta x), \tag{4.23}$$

$$u_i^{n+1} - u_i^n = \Delta t F(u^{n+1}, \Delta x). \tag{4.24}$$

Let us denote the *truncation error* or the *local discretization error* as $R_i = Du_i - Lu_i$, , where $Du = F(u)$ is the spatial differential operator and L is the difference operator that approximates derivatives in the operator D. A difference operator L is termed *consistent* with the differential operator D, if the the norm of the truncation error tends to zero as the grid size goes to zero, i.e. $\|R\| \to 0$ as $\Delta x \to 0$. We see that the consistensy is a property of operator L and not the solution's property. To analyse the solution we should introduce a concept of the *global discretization error* as $\epsilon_i = y_i - u_i$, where y_i is the exact solution value of differential equation and u_i is the approximation at grid point i. A difference operator L is *convergent* to the differential operator D if $\|\epsilon\| \to 0$ as $\Delta x \to 0$ [3].

The TPFA method is commonly used in practical reservoir simulation. A good advantage of this method is its simplicity and computational speed. However the TPFA disretisation requires K-orthogonal grid and that means the parallel alignment of $\boldsymbol{K}\vec{n}_{i,k}$ and $\vec{d}_{i,k}$ in formula (4.6). In the case of non-orthogonal grid the TPFA discretization is not consistent [21] and for such cases we do not expect the numerical solution to converge to the true solution as the spatial grid is refined.

Consider the explicit method (4.23) for conductive part, when $F(u) = \eta u_{xx}$. When such scheme is disretized with finite difference method [3], [2], then such discretization leads to stability condition on the temporal step

$$\frac{\eta \Delta t}{(\Delta x)^2} \leq \frac{1}{2}, \tag{4.25}$$

where $\eta = \boldsymbol{K}/(\rho c_p)_{eff}$. In case of advective term, when $F(u) = vu_x$, Courant-Friedrics-Lewy condition that should be fulfilled is as follows

$$\frac{v \Delta t}{\Delta x} \leq 1, \tag{4.26}$$

where v is the wavespeed or the magnitude of velocity [2]. In other words formula (4.26) implies the flow front to move no more than one grid block per time step. In our scheme we use these conditions on a time step in accordance with predominant term in the energy equation. The implicit scheme (4.24) is unconditionally stable, however large time steps can lead to numerical diffusion [27].

Chapter 5

Coarse Scale Methods

Fine scale models of reservoirs in our further simulations contain 13200 grid cells and this large number generates large matrices in the solvers. Thus the energy equation solvers need sufficient computational capacity to process such detailed model in a short time. There are two currently available ways to reduce the computational time: to decrease the number of grid cells by upgridding-upscaling procedure or to develop more efficient simulators [16]. In this chapter we are going to focus on upscaling of reservoir properties in relation to solving the energy equation. Upscaling is closely connected to upgridding of a geological grid. *Upgridding* is a process of creating a coarse grid and *upscaling* of reservoir parameters is a process of obtaining values on the upgridded cells from the fine grid.

5.1 Upscaling

A great number of different techniques are developed to upscale reservoir parameters and in different situations different upscaling procedures are appropriate. Here we will describe those upscaling methods that are suitable for our model. At first we consider types of parameters that are used in our equation. We first solve the pressure equation for a single-phase flow on a fine grid and obtain fine scaled fluxes and pressures. At this stage we should upscale porosity and fluid velocities. Afterwards we solve the energy equation (3.7) on a coarse grid using pre-computed upscaled parameters: velocity and porosity. Heat properties are

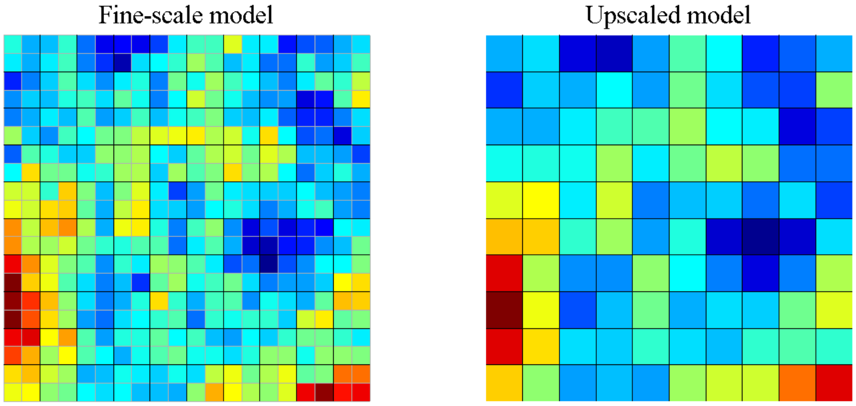


Figure 5.1: Volume averaging of porosity.

another sort of parameters to be upscaled in our model. Fluid and solid properties are involved in the energy equation and thus we should upscale heat capacities and thermal conductivities for different materials.

We will compute porosity on a coarse scale ϕ^* as follows

$$\phi_B^* = \frac{1}{V_B} \int_{V_B} \phi dV \approx \left(\sum_{i \in B} \phi_i V_i \right) / V_B, \tag{5.1}$$

where the subscript B designates a coarse block and V_B is a bulk volume of block B, [9]. The example of upscaled porosity is shown in Figure 5.1. We upscale porosity from a 20×20 fine scale grid to a Cartesian grid of 10×10 cells by taking a mean value on every coarse cell. Here the right figure shows fine grid cells that are averaged within coarse cells and we see that many details are lost.

The coarse scale flow rate q^* [m^3/s] across the interface l can be found from the formula below

$$q_l^* = \sum_{i=1}^{N_l^f} q_i, \tag{5.2}$$

$$q_i = A_i \vec{v}_i \cdot \vec{n}_i, \tag{5.3}$$

where N_l^f is the number of fine faces on a coarse face l, A is a face area [m^2], $\vec{n}$ is an outward normal and $\vec{v}$ is velocity vector [m/s]. Normals on a coarse grid face can be found as the sum of fine scale normal vectors on this coarse face.

In our simulations we assume isotropic heat capacity for both fluid and solid and thus we can set a value at any fine grid cell as a coarse cell value. In general case since the heat capacity does not depend on grid cell volumes we should use simple arithmetic mean as an upscaling technique

$$(c_p)^*_B = \sum_{i=1}^{N^c_B} (c_p)_i / N^c_B, \tag{5.4}$$

where N^c_B is the number of fine scale cells in a block B.

To upscale an effective thermal conductivity we can apply various techniques. The simplest one is an average upscaling of fine cell conductivities to a coarse scale. However such upscaled permeability would not generate the same flow rate as a fine scale computation. To achieve more precise solution we should use some other methods like power averaging procedures [9], [7] or flow-based upscaling. Such techniques are usually applied to permeability upscaling, and since conductivity tensor plays the same role in the conductive term as permeability in the pressure equation (2.12) we have good reasons for utilizing these methods in conductivity upsaling. In this work the flow-based upscaling is used to scale up effective conductivity and such method is significant when the coarse grid geometry is not K-orthogonal. The main principle of this upscaling procedure is to find fluxes at coarse boundaries and then compute the thermal conductivity values using Darcy's and Fourier's law [8]. The conductivity value in x direction for a coarse grid cell can be computed via

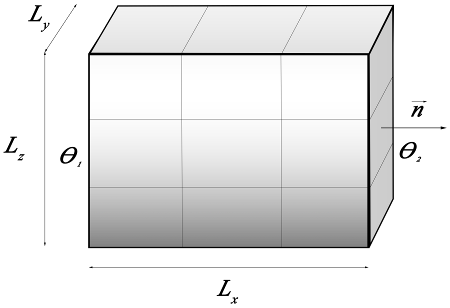


Figure 5.2: Coarse grid block.

$$\boldsymbol{k}^*_x = \frac{\psi_x L_x}{A(\Theta_2 - \Theta_1)}, \tag{5.5}$$

$$\psi_x = \int_0^{L_y} \int_0^{L_z} \vec{\psi}(L_1, y, z) \cdot \vec{n} dz dy, \tag{5.6}$$

where ψ_x is the heat rate through the right boundary, L_x is the length between two boundaries in x direction, $A = L_y \cdot L_z$ is the surface area, Θ_1 and Θ_2 are the temperatures at the left and right boundaries (see Figure 5.2). The same way we can upscale conductivities in y and z directions. Afterwards upscaled effective conductivity tensor is used in formulas (4.6) and (4.8) to calculate transmissibility on a coarse grid.

5.2 Upgridding

The spatial resolution of a coarse grid model is determined by upgridding techniques and often the accuracy of an upscaled model can be improved by an adequate grid. There are many various types of grids that are utilized in different models [9], but we turn our attention to simple uniform partitioning of a structured grid (Cartesian grid) and flow-based coarsening based on amalgamation that was proposed in [1], [18], [17]. The example of Cartesian grid was introduced in the previous chapter and it is illustrated in Figure 5.1. In this example a single coarse grid cell incorporates four underlying fine grid cells and coarse grid lines are aligned with fine grid edges. This grid type is widely known and may be used in various methods including FVM and FDM. Nonuniform coarse grids are usually utilized when it is necessary to reach accuracy in some local areas of reservoir. For instance some flow-based grids introduce high level of resolution in regions of high flow and coarse resolutions when the flow is low.

A flow-based nonuniform coarsening algorithm is used to generate coarse grids for transport simulations and we are going to apply this algorithm for heat transfer simulations. Such coarse scale grid is very effective for advective processes in highly heterogeneous reservoirs, because these coarse blocks match better to the flow patterns than simple structured grids. Also it was found that such grids are robust to changing flow conditions and that is why it is not necessary to regenerate the grid [1], [16]. The algorithm utilizes fine scale components referred to as indicators. Indicators can be based on velocity, permeability, time-of-flight or some other flow parameters. The example of coarse grids based on different indicators is shown in Figure 5.3.

We create indicator functions as a logarithmic scaling of chosen fine scale parameter y

$$g(y) = \log|y| - \min(\log|y|) + 1. \tag{5.7}$$

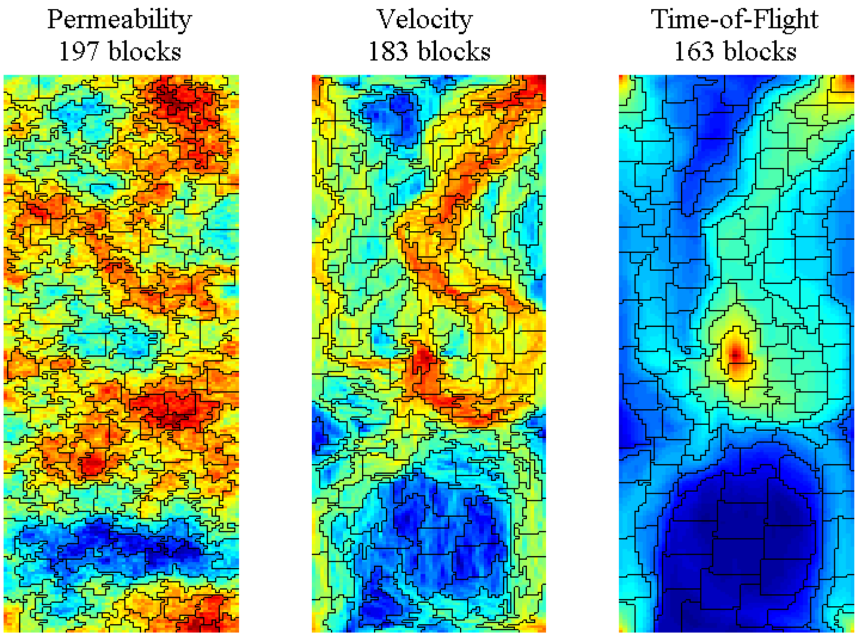


Figure 5.3: The resulting coarse grid for different indicators.

The flow-based coarsening algorithm consists of four steps, [16], that are shown in Figure 5.4:

1. Generating an initial partition according to logarithm of chosen indicator g.
2. Merging of blocks smaller then lower bound on volume according to a prescribed indicator function.
3. Refining of blocks that have too large amount of flow according to a prescribed indicator function.
4. A final merging of blocks (the step 2 is repeated).

Before we start the algorithm, we should define a number of bins, which are utilized in partition generating in the first step, a lower bound on cell volumes, N_L, that will prevent the algorithm from generating too small blocks and an upper bound on total amount of flow through each grid block, N_U, that prevents from generating too large blocks. The important requirement of the algorithm

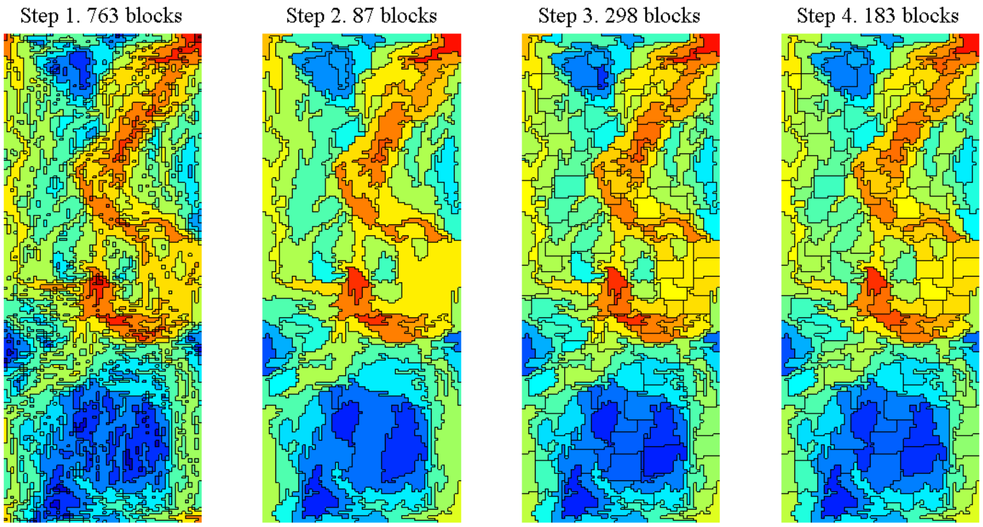


Figure 5.4: Four steps of the flow-based coarsening algorithm.

is that the coarse grid cells should be connected and they should separate high-flow regions from low-flow regions. After a first step we obtain a coarse grid with many small blocks and this problem is solved by merging small cells with neighbouring cells in the second step. After merging the flow through some of new cells is above the upper flow bound N_U and then the refinement algorithm is applied. For a coarse block B with too high flow an arbitrary fine grid cell c_0 on a boundary ∂B is picked. Then furthest fine grid cell $c_1 \in B$ from cell c_0 is defined as a separate block B' and this new block is enlarged with surrounding cells from block B until the flow through the block becomes lower than N_U. Then the step 2 is repeated for the reason that some blocks violate the lower bound. The last step reduces the number of blocks by 30-50% [16].

Chapter 6

Numerical Experiments

In this chapter we will perform simulations of mathematical models described in Chapters 2 and 3 by using numerical schemes from Chapter 4. All source code is written in MATLAB (MATrix LABoratory), which is widely known programming language for technical computing. The resulting figures that are presented in this work have been created in MATLAB with the code that is based on the MATLAB Reservoir Simulation Toolbox (MRST), which is an open-source software and was developed by SINTEF [31] especially for reservoir simulation. An important property of MRST is the ability to simulate models on different types of grids, both structured and unstructured, because every grid is determined by fields that contain information about cells, faces and nodes. The separate structures hold information about rock, fluid, reservoir, sources, wells and boundary conditions, and one should initialize all these structures prior to solving a problem.

The basic pressure and transport solvers for incompressible, single- and two-phase flow are contained in the core part of MRST. Also MRST consists of various add-on modules where we can find specialised functions and a range of different solvers. In our code we use the 'spe10' module for simulations in a heterogeneous medium. The 'coarsegrid' and 'gridtools' modules are used to generate different types of coarse grids and to work with grid properties. The 'upscaling' module contains various functions for a flow-based upscaling of permeabilities. And finally we need the coarsening by amalgamation module ('agglom'), which was developed for constructing coarse grids based on amalgamation of fine grid cells. MRST does not hold any energy equation solvers and thus we create our own module 'heattransfer' with the explicit and implicit solvers for a linear convection-conduction equation that are compatible with any type of grid in MRST. The solvers in this module can simulate the conservation of energy equation and return reservoir solution with the vector of temperatures in every grid cell as a result.

6.1 Fluid Flow Model

6.1.1 Homogeneous medium

Here we want to implement the pressure equation (2.12) in a homogeneous medium when parameters of the system are spatially uniform. We assume that a reservoir layer is thin enough to neglect gravity forces, and thus we can regard our model as 2D. Consider 50 × 50 Cartesian grid with physical dimensions of 100 × 100 meters, uniform porosity of 0.3 and isotropic (not depending on the direction) permeability of $[10, 10]$ mD. In MRST porosity and permeability values are set for every grid block as rock structure field. We define the fluid properties next and take the density as $\rho = 1$ kg/m^3 and the viscosity as $\mu = 1$ Poise. Let the fluid flow in through the left boundary and assume constant pressure on the right boundary, so we have both Dirichlet (2.13) and Neumann (2.14) conditions in the system and absence of sources or sinks, i.e. source term is equal zero in every grid cell. In Figure 6.1 we can see the solution of the pressure equation (2.12). Obviously, pressure values are higher nearby injection boundary and lower close to the right boundary, where the pressure is fixed. From Darcy's law (2.6) velocity values can be found and time-of-flight is calculated with formula (2.19).

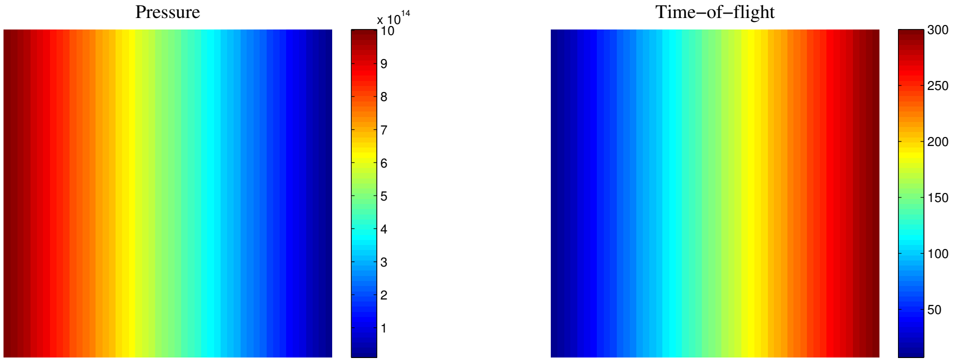


Figure 6.1: Pressure [Pa] and time-of-flight [s] for a homogeneous medium.

6.1.2 10th SPE Model

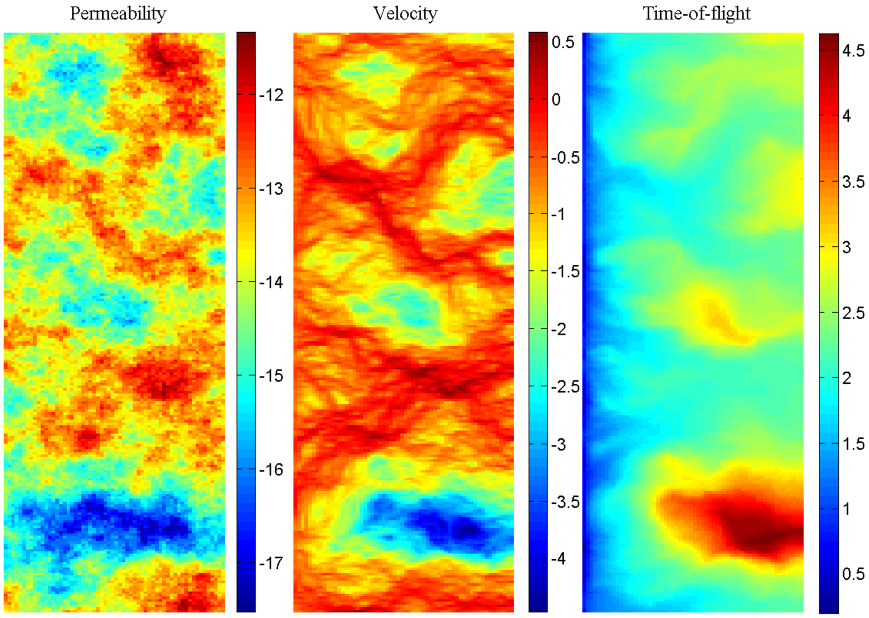


Figure 6.2: Common logarithm of permeability, velocity and time-of-flight for the boundary flux problem on the 1st SPE 10 layer.

Here we use rock and fluid parameters from the SPE Comparative Solution Project model 2 [33]. SPE 10 is used for a comparison of different computational and upscaling methods because of its high degree of heterogeneity. This model is determined by Cartesian grid 60 × 220 × 85 with the dimensions of 1200 × 2200 × 170 ft.

In this example we set density as $\rho = 1014$ kg/m^3 and viscosity as $\mu = 1$ cP. The rock properties porosity and permeability are heterogeneous and depend on a layer of the model. Porosity is in range 0.1 - 0.4 in SPE 10 model and such value corresponds to natural rock values. In MRST permeability is defined by $N \times 3$ matrix $(K_1(i), K_2(i), K_3(i))$, $i = 1, ..., N$, and then permeability tensor is equal to

$$K_i = \begin{bmatrix} K_1(i) & K_2(i) \\ K_2(i) & K_3(i) \end{bmatrix}.$$

For this medium we consider the same setup as in the previous example. Fluid flows into the reservoir through the left boundary and pressure is fixed on the right boundary. We solve the pressure equation (2.12) with such boundary conditions in the heterogeneous medium from the first layer of SPE 10 model. In Figure 6.2 we can see common logarithm of the permeability on this layer, the resulting velocity and time-of-flight. It can be noticed how velocity and thus time-of-flight depend on the permeability distribution. In regions, where permeability is low, the fluid velocity values are also small and more time is needed for fluid to reach such areas, thus time-of-flight values are higher in these cells.

6.2 Heat Transport Model

6.2.1 Boundary flux problem in a homogeneous medium

In the first experiment we consider homogeneous medium that was described before in section 6.1.1. In this example a fluid flows in through the left boundary of the reservoir and flows out through the right boundary. Let us set the initial temperature of reservoir as 30℃, that is equal to 303.15 K, and the injection fluid temperature equal to 20℃ or 293.15 K. Also we assume no heat flow through other boundaries or, in other words, we assume homogeneous Neumann boundary conditions. Next we should set specific heat capacity and thermal conductivity values for both fluid and solid. In this experiment for simplicity we set fluid and solid heat capacities as 1 J/(kg· K) and thermal conductivities are equal to 1 W/(m· K).

Table 6.1: Euler's forward method's computational time.

Number of steps	CPU time [s]
251	0.58
500	1.069
1001	2.029
2001	4.031
4000	7.771

To solve the energy conservation equation (3.7) we use two time-stepping methods: Euler's forward scheme (4.21) or Euler's backward scheme (4.22). It is known that the explicit scheme is computationally faster, but can be numerically unstable. On the contrary, the implicit method has no requirements on time-step size, but needs more computational time and numerical diffusion can appear for a large time step when the convective term dominates in the convective-diffusive equation. The computational time for Euler's forward method for different number of time step is shown in Table 6.1. The results in Table 6.1 correspond to the final time $T = 50$ s and this value is chosen in accordance with other model parameters in order to perform the explicit and implicit schemes' capabilities. For a real heat transfer simulation the final time is bigger by many times.

Here and further we measure the overall accuracy of the numerical solution u as the L_2 norm of difference between u and the reference solution u_{ref}, divided by the L_2 norm of the reference solution

$$\varepsilon(t) = \frac{\|u(t) - u_{ref}(t)\|_{L_2(\Omega)}}{\|u_{ref}(t)\|_{L_2(\Omega)}}. \tag{6.1}$$

To find Euler's backward method's accuracy we use an explicit solution with 4000 time steps as the reference solution in formula (6.1). The spatial discretization for the explicit and implicit scheme is the same and we change only the temporal discretization of the implicit scheme. The error and computational time in seconds for the implicit scheme is shown in Table 6.2. We can notice that the computational time for the implicit scheme is larger. However in further computations the explicit scheme requires too many time steps for convergence and in many cases it becomes infeasible to use Euler's forward scheme, because the total computational time becomes too large. According to the error from Table 6.2 we can afford to utilize Euler's backward scheme in our further simulations and so reduce the total computational time.

Table 6.2: Euler's backward method's computational time and error

Number of steps	CPU time [s]	Error ε
10	0.339	0.0258
20	0.545	0.0151
50	1.220	0.0069
100	2.307	0.0036
500	11.222	0.000828

The error for the implicit scheme at logarithmic scales of axes is shown in Figure 6.3. We can see in the figure that the plot of the error is parallel to the function $f(x) = 1/x$ and this means that we obtained the 1st order convergence rate as was expected.

In Figure 6.4 we can see how the cold temperature front passes through the reservoir from left to right. In these examples we used Euler's forward scheme to compute the solution. Figures in the left column illustrate the heat flow, when the heat transfer by motion of fluid (advection) has stronger influence on the behaviour of a heat expansion than molecular motion (conduction). In this case

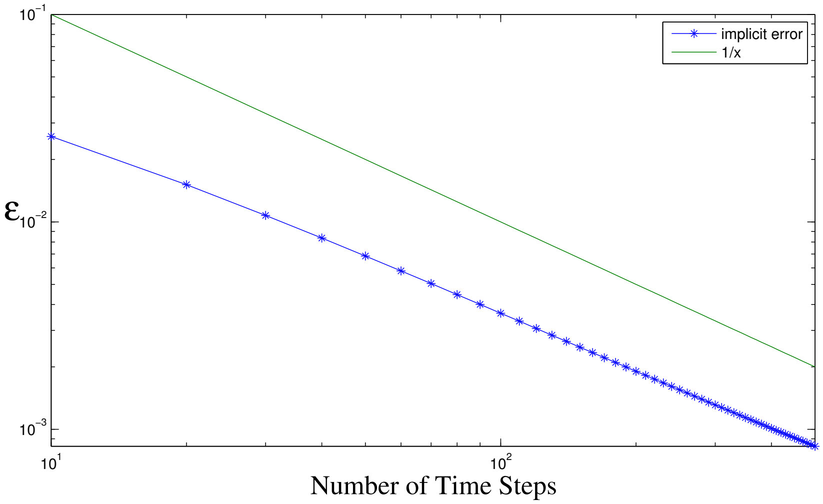


Figure 6.3: Error of the implicit scheme with various numbers of time steps for the boundary flux problem and plot of the function $f(x) = 1/x$ at logarithmic scales of axes.

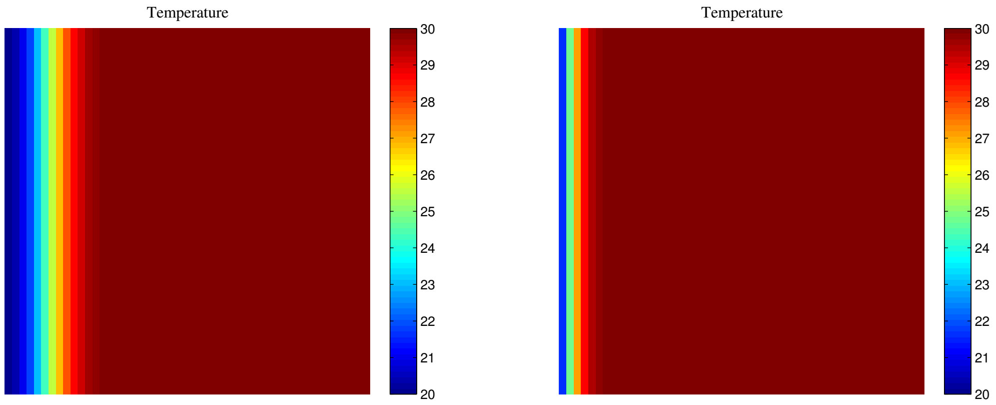
we set velocity of a fluid equal to 10 m^3/s and Péclet number equals 2. The right column on the contrary shows solution of the energy equation when conduction dominates over convection, here the velocity is equal to 10^{-1} m^3/s and Péclet number equals to 0.02. To compute Péclet number we used the formula (3.8) for every grid cell

$$Pe_i = \frac{L_i \vec{v}_i \rho c_p}{\boldsymbol{k}_i}$$

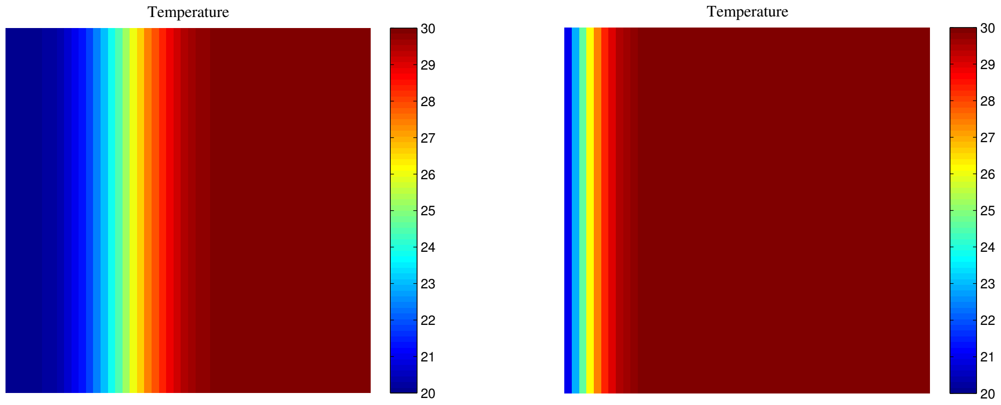
with the characteristic length L_i of square root of the grid cell's area. The conductive term in the equation (3.7) dominates when Péclet number is less than 1. However in different models conductive/convective dominated region can be slightly shifted towards Péclet number's scale due to the choice of the characteristic length.

6.2.2 Boundary flux problem in a heterogeneous medium

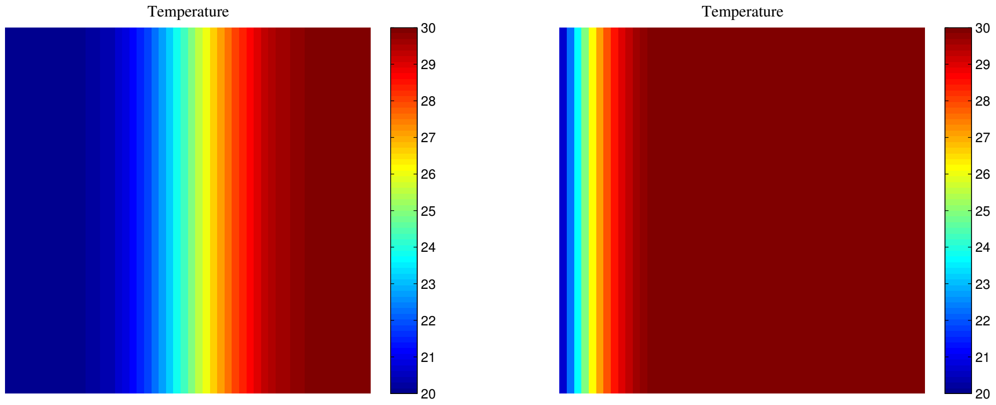
In this example we consider a problem on the 1st layer of SPE 10 heterogeneous medium. As before consider fixed flux through the left boundary and fixed pressure on the right boundary of the domain. We set initial temperature equal 30℃ in every grid cell and the injection flux temperature equal 20℃. According to the Tables 3.1 and 3.2 we assign fluid specific heat capacity as $(c_p)_f = 4 \cdot 10^3$



(a) Time = 10 s (b) Time = 10 s



(c) Time = 30 s (d) Time = 30 s



(e) Time = 50 s (f) Time = 50 s

Figure 6.4: Temperature distribution in reservoir for different final time. Plots (a), (c), (e) correspond to Péclet number of 2 and plots (b), (d), (f) correspond to low Péclet number of 0.02.

J/(kg·K) and fluid thermal conductivity as $\boldsymbol{k}_f = 0.58$ W/(m· K) that correspond to water properties. The solid heat capacity and thermal conductivity we set as $(c_p)_s = 0.9 \cdot 10^3$ J/(kg·K) and $\boldsymbol{k}_s = 2$ W/(m· K). The last parameter to assign is solid density, which we assume equal to constant 2600 kg/m^3. Such properties are quite close to real physical values. The heat parameters for fluid and solid are defined in Table 6.3 and correspond to natural properties. We start our simulations by solving the energy equation with boundary velocity equal to 50 m^3/s and final time $T = 500$. With such values average Péclet number for grid cells equals $1.43 \cdot 10^6$, the maximum value of Péclet numbers is $2.137 \cdot 10^7$. This corresponds to the situation, when advection strongly dominates over conduction.

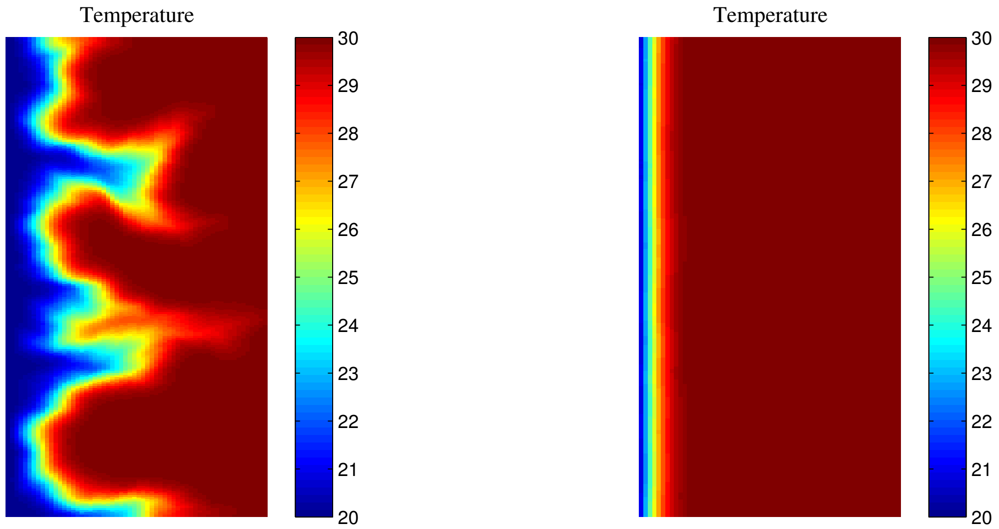
Table 6.3: Heat parameters in experiments.

Property	Value	Unit
$(c_p)_f$	$4 \cdot 10^3$	J/(kg·K)
$(c_p)_s$	$0.9 \cdot 10^3$	J/(kg·K)
$\boldsymbol{k}_f$	0.58	W/(m· K)
$\boldsymbol{k}_s$	2	W/(m· K)
ρ_s	2600	kg/m^3
ρ_f	1014	kg/m^3
μ	1	cP

We first solve the equation (3.7) with the explicit method (4.23) with different number of time steps. In Table 6.4 the dependency between computational time and number of time steps is evident. In this example the method converges when Δt is quite large, however in numerous cases the explicit method requires around 10^6 time steps and the computational time becomes too long and can take many hours or even days. Then it is a better alternative to use the implicit scheme (4.24), which has no requirements on convergence and thus is much faster. In Table 6.5 the error (6.1) for the implicit scheme is shown, where the solution with 4000 time steps computed by the explicit method works as reference solution. The error reduces according to decrease of time step Δt with the 1st order convergence rate as in the previous example.

Table 6.4: Euler's forward method's computational time.

Number of steps	CPU time [s]
500	5.227
1000	9.852
2000	19.535
4000	36.209



(a) $\vec{v} = 5 \cdot 10^{-5}$ m^3/s (b) $\vec{v} = 5 \cdot 10^{-7}$ m^3/s

Figure 6.5: Temperature distribution for two problems with different velocities.

In Figure 6.5 we can see solutions of two problems with different Péclet numbers and final time $T = 5 \cdot 10^8$ s. For the first problem we chose injection velocity equal to $5 \cdot 10^{-5}$ m^3/s, then Péclet number is 21.37 and advection is predominant term in the equation. Here the location of the temperature front depends on velocity field and looks curved in figure (a). The temperature front in figure (b) is straight, because it has 10^2 times smaller influence of the velocity and high influence of effective thermal conductivity ($Pe = 0.21$), which is much more regular.

Table 6.5: Euler's backward method's computational time and error.

Number of steps	CPU time [s]	Error ε
10	2.205	0.0083
20	3.322	0.0045
50	7.418	0.0019
100	14.074	0.0010
500	68.710	0.00022

6.2.3 Quarter-five spot problem and numerical diffusion.

In this experiment we consider a quarter-five spot problem for a reservoir with heterogeneous parameters from the 1st layer of the SPE 10 model, where source is in the lower left corner and sink is in the upper right corner of the reservoir. In

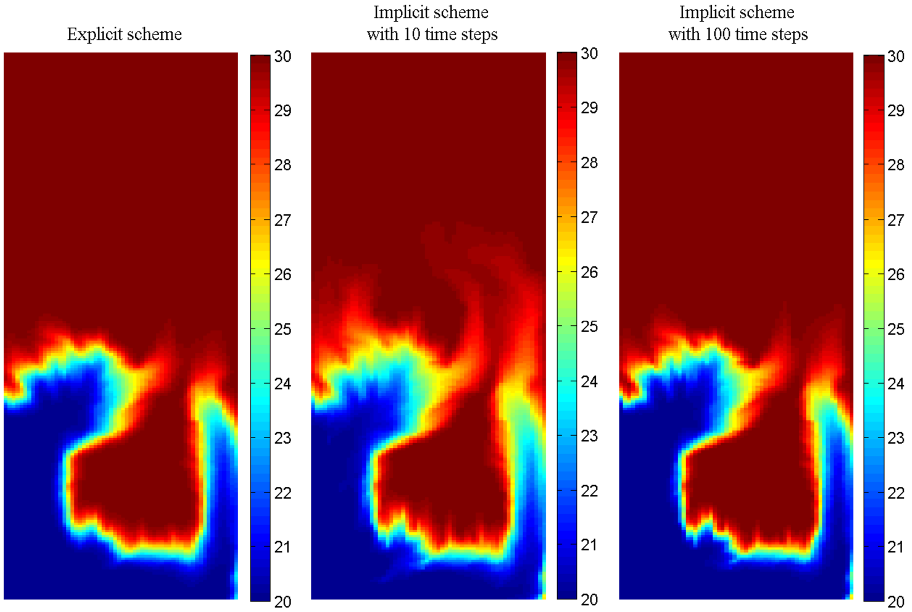


Figure 6.6: Numerical diffusion effect in the implicit scheme with different number of time steps in comparison to the explicit scheme.

equation (2.12) we set non zero source value Q in the first and last grid cells. We assume zero fluid and heat flow through the boundaries of the domain. Let us set the initial reservoir temperature to 30℃ and injection temperature to 20℃. Also we assume zero heat flow through all domain boundaries. Heat properties are defined as in the previous example and are listed in Table 6.3. For this problem Euler's forward method (4.23) requires quite small time step value Δt and thus the computational time is large. CPU time for explicit scheme is listed in the Table 6.6 and it seems reasonable to use Euler's backward scheme (4.24) in further simulations.

Table 6.6: Euler's forward method's computational time

Number of steps	CPU time [s]
4415	43.751
8829	89.190
17658	171.136

However when the convection term dominates in the energy conservation equation one should remember about numerical diffusion. In Figure 6.6 the results of the implicit methods with different time steps are presented in problem with Péclet number of $3.48 \cdot 10^8$, when the dominance of convection is much stronger than conduction. The left plot shows explicit solution, the plot in the middle illustrates the numerical diffusion of implicit method with 10 time steps, and the last plot shows the solution of implicit method with 100 time steps. We can notice large numerical error in the second figure and less error in the third figure that is explained by numerical diffusion, which always appears in methods, where the upwind scheme is used and the time step is large [27]. The errors of the implicit scheme are listed in Table 6.7. As an example, we calculated the error in the experiment with less influence of convective part, when Péclet number is equal to 0.5, and the error for this problem is shown in Table 6.8. We can see from the table that the error is smaller in this case and thus the numerical diffusion is reduced to minimum. The explicit scheme in this case requires more then 10^4 steps and obviously the implicit scheme is much faster. So with several experiments we found that the numerical diffusion effect appears in the implicit scheme for the convective-dominated energy equation (3.7). In further simulations to achieve more accurate solution we should increase the number of time steps according to the Péclet number, when advection dominates in the energy conservation equation.

Table 6.7: Euler's backward method's computational time and error for problem with high Péclet number.

Number of steps	CPU time [s]	Error ε
10	2.046	0.0191
20	2.895	0.0118
50	7.018	0.0057
100	14.050	0.0031
500	68.960	0.000698

Table 6.8: Euler's backward method's computational time and error for problem with low Péclet number

Number of steps	CPU time [s]	Error ε
10	1.681	0.000473
20	2.762	0.000239
50	6.961	0.000096
100	13.962	0.000048
500	68.067	0.000001

6.3 Numerical Experiments on a Coarse Scale

6.3.1 Homogeneous medium

Here we will run experiments of a boundary flux problem in a homogeneous medium. Now we scale the heat and fluid parameters to 1 except the thermal conductivity. We are going to vary thermal conductivities in every simulation and thus we will obtain results at different Péclet numbers. Our simulations start with large thermal conductivity value of 10^3 W/(m· K) and Péclet number of $2 \cdot 10^{-4}$, which means that conduction strongly dominates over convection in the energy equation (3.7). Then we will reduce thermal conductivity until only convection term will influence on the heat transfer. Also we will compute total reservoir temperature and check if it also has influence on the resulting error. For instance, when the fluid conductivity is too high, the temperature in reservoir will tend to minimum very fast and thus the error tends to zero.

According to our previous conclusions we will use the implicit scheme in the coming simulations. We first choose indicators and compute indicator functions with formula (5.7). The lower and upper bounds on a coarse block from the flow-based coarsening algorithm are $N_L = 2$ and $N_U = 5$ respectively. We will compare grids based on already known indicators like permeability, velocity and time-of-flight [16] with the grid based on effective thermal conductivity as indicator and with a uniform Cartesian grid 25×25. Also we tried to use Péclet number as indicator and it was found out that such type of indicator gives the same grid structure as velocity indicator, because of straight dependency in formula (3.8). Therefore we do not use it in our simulations. Then we apply a flow-based coarsening algorithm to obtain coarse grids according to the chosen indicators. In the refinement step we use 'refineGreedy2' method and it has similar error as other types of refinement algorithms referring to [24]. In Figure 6.7 solutions on different grids are shown. As we can notice, grids that are based on uniform permeability and conductivity have the same structure as the Cartesian grid for chosen N_L and N_U. In the first plot we have solution for a problem with low thermal conductivity of 10^{-3} W/(m· K) and Péclet number of 200. The second plot shows the result for predominant conductive term, when thermal conductivity equals to 10 W/(m· K) and Péclet number is 0.02 .

The errors for the solutions with different Péclet numbers are shown in Figure 6.8. The error ε is computed with formula (6.1), where the fine grid solution computed by the implicit scheme with 100 time steps works as reference. In the

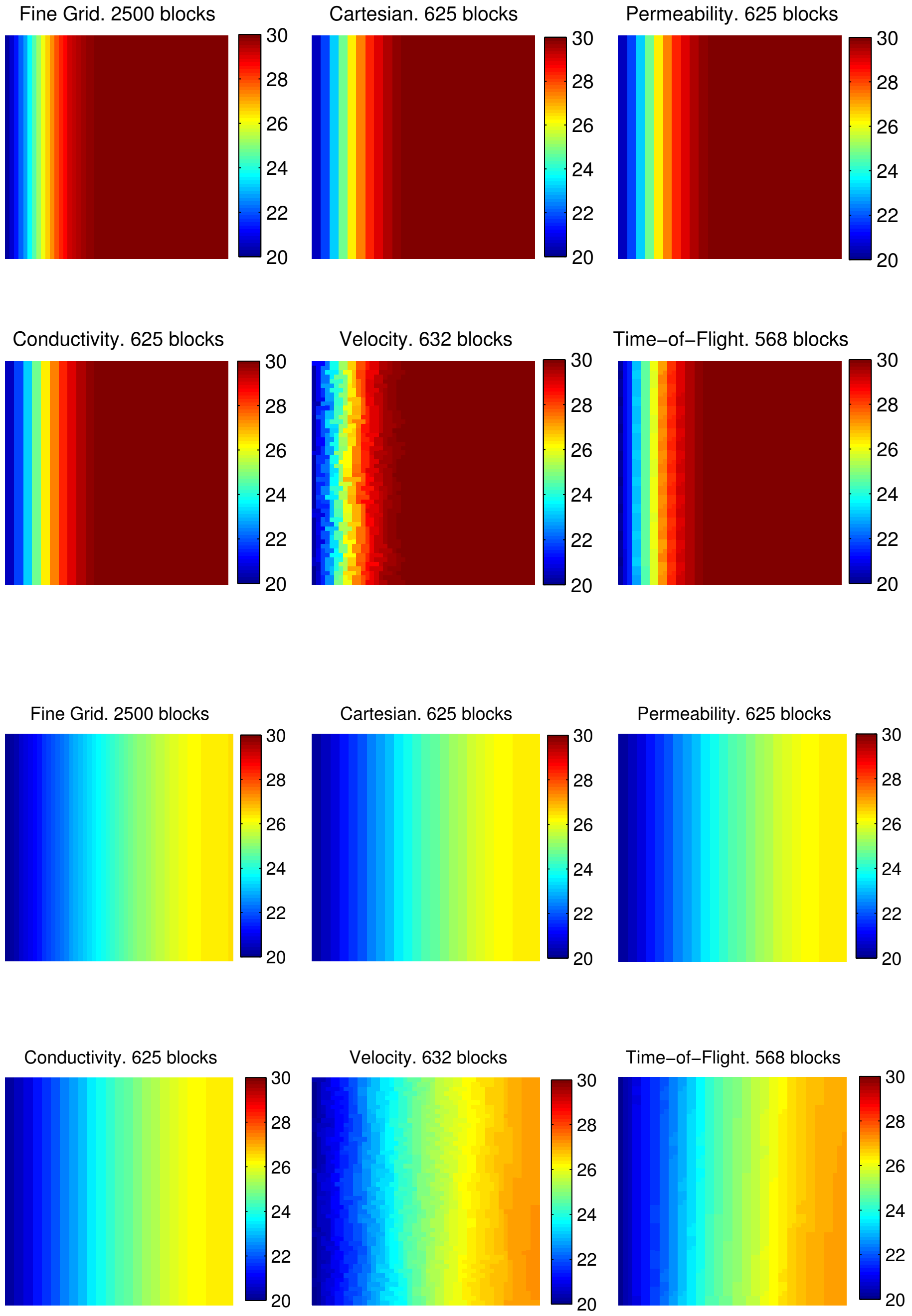


Figure 6.7: Simulations of the energy equation with two different Péclet numbers of 200 and 0.02 on various grids in a homogeneous medium.

left part of the figure error tends to zero while total reservoir temperature is close to injection temperature of 20℃. Also we can see large error in the area where conductive part dominates. It is reasonable to assume that this error appears due to inconsistency of the two-point flux approximation discretization of the conductive term. TPFA method does not converge spatially to the true solution on non K-orthogonal grids and thus for small Péclet numbers the accuracy error is quite large. The error tends to constant with the decrease of thermal conductivity and here fluid motion has big influence on the heat transfer. In this case flow-based coarse grids should be very effective. We see in Figure 6.8 that the time-of-flight coarse grid has minimal error with low number of grid cells and velocity indicator gives similar accuracy as Cartesian grid. Nonuniform grids consist of various irregularities that have a big influence on half-transmissibilities calculation (4.6) and thus transmissibility matrix. Hence for methods with TPFA implementation the solution on K-orthogonal grids is the most accurate. So we can conclude that Cartesian grid is a good choice for a boundary flux problem with a low Péclet number.

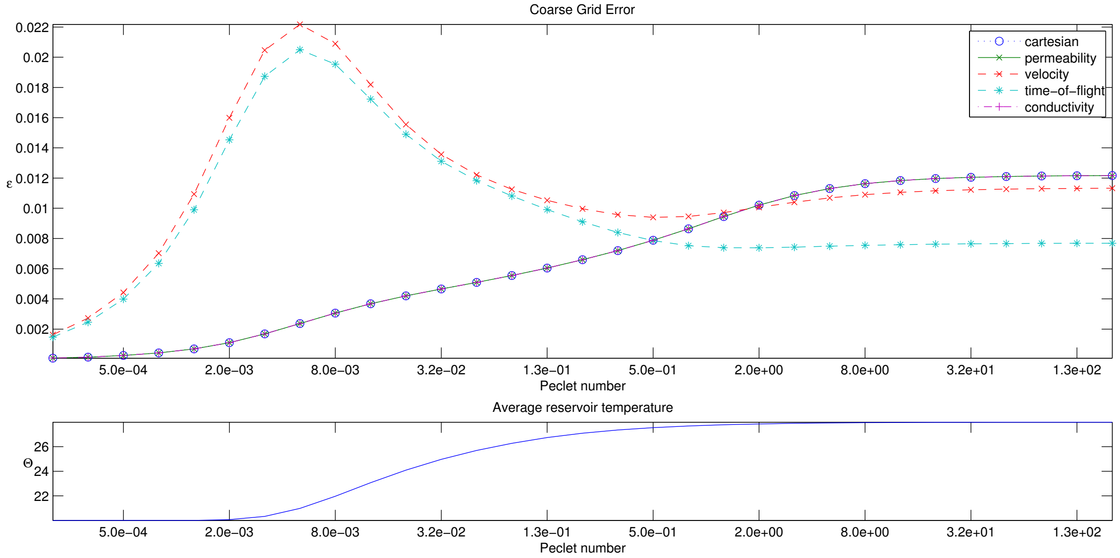
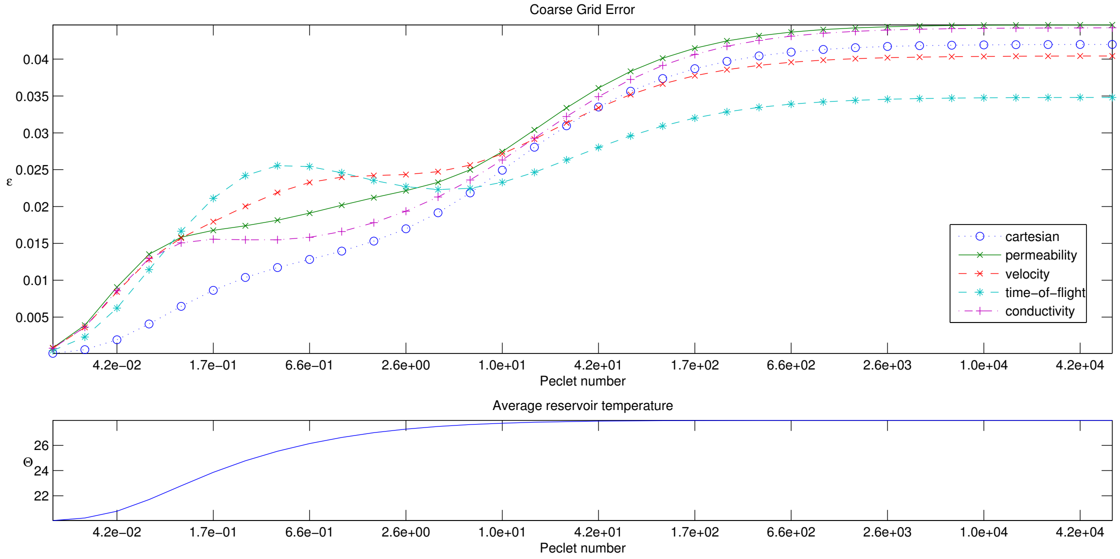


Figure 6.8: The error and average reservoir temperature for various Péclet numbers on a homogeneous medium.

6.3.2 Heterogeneous medium

In this section we will run a number of experiments in the heterogeneous medium from the 1st layer in the SPE 10 model. The permeability field of the 1st layer


Coarse Grid Error

Average reservoir temperature

Figure 6.9: The error and average reservoir temperature for various Péclet numbers on the 1st SPE 10 layer heterogeneous medium.

is shown in Figure 6.2. We set heat and flow parameters as in section 6.2.2 in Table 6.3. In this experiment we run a number of simulations with different thermal conductivity value for the problem from section 6.2.2. As before we start simulations at low Péclet number of 0.0165, when conductive term dominates, and then we decrease effective thermal conductivity value and so increase the dominance of advective term. For each simulation we compute the error with formula (6.1) and build a plot to show the error dependency from Péclet number.

As in the previous section we compare results on different grids. Flow-based upgridding is of our interest and we will compare grids based on various indicators like permeability, conductivity, time-of-flight and velocity. In this numerical experiment we choose the lower bound on cell volumes $N_L = 15$ and the upper bound on total amount of flow through coarse cell $N_U = 30$. These bounds prevent the coarsening algorithm from generating too small and too large blocks. The Cartesian grid 10×40 works as a benchmark of standard gridding in our simulations. Uniform coarsening should give us the smallest error for the two-point flux approximation scheme due to its K-orthogonality. In Figure 6.9 we can see that the total temperature of reservoir tends to zero when thermal conductivity value is too large, i.e. Péclet number is low. We will not consider results with too low total temperature in the reservoir.

In the interval, when Péclet numbers are less than 10, we can see that flow-based

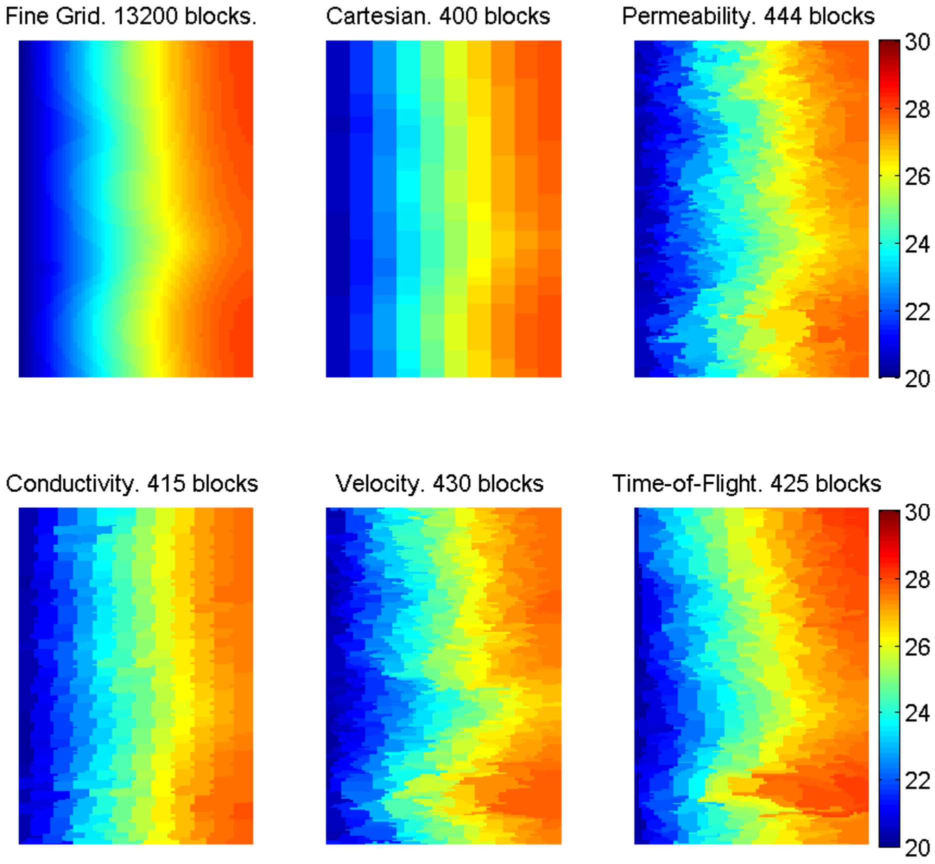


Figure 6.10: Simulations of the conductive-dominated energy equation on various grids on the 1st SPE 10 layer.

grids have less accurate results in comparison with simple Cartesian grid. Such behaviour may occur due to inconsistency of the two-point flux approximation scheme on nonuniform grids or due to irregular coarse grid cells' shape. The solution at Péclet number of 0.26 and $\boldsymbol{k}_{eff} = 2.511 \cdot 10^7$ W/(m$\cdot$ K) is shown in Figure 6.10 and we can see the evident dominance of conduction here. It is obvious from the figure that flow-based grids have less accuracy. Heat front moves almost without variations along x direction on fine and Cartesian grids, but on nonuniform grids it looks irregular. The Cartesian grid and the coarse grid that is based on conductivity have similar behaviour, because the fine scale thermal conductivity value is uniform and the coarsening algorithm generates grid with more regular cells then in case of heterogeneous indicators. Permeability, velocity and time-of-flight indicator based grids consist of many extended cells along the

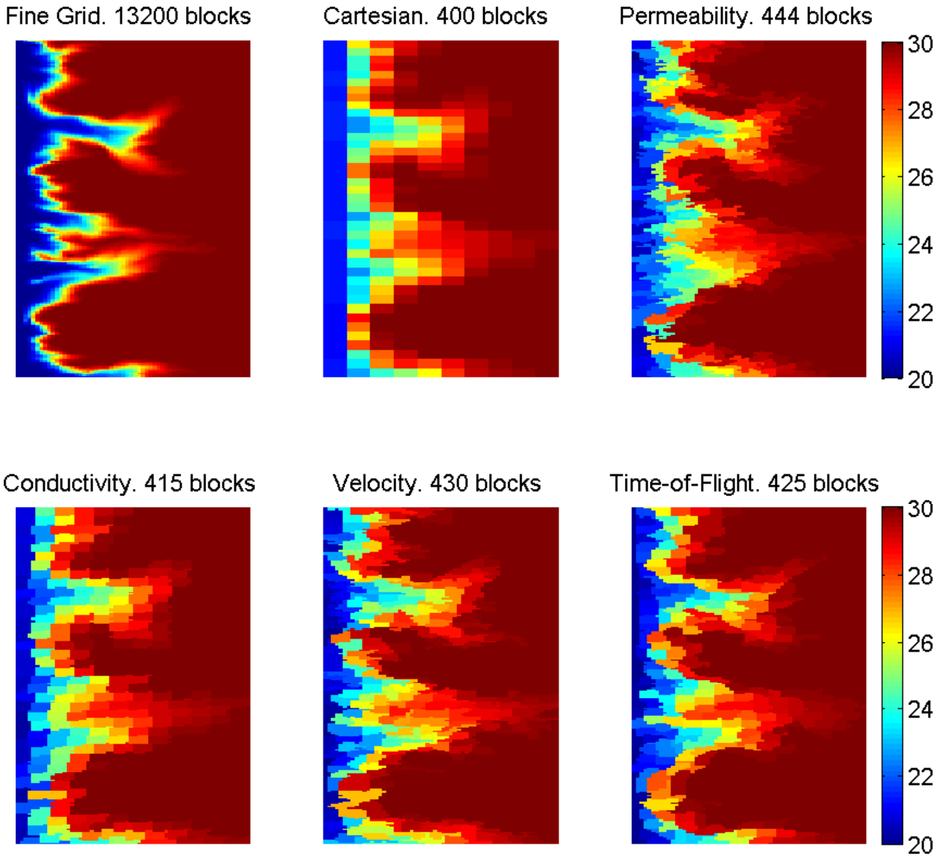


Figure 6.11: Simulations of the convective-dominated energy equation on various grids on the 1st SPE 10 layer.

fluid flow and thus such grid structures can strongly influence the accuracy. It seems that for such kind of problem with boundary inflow and low Péclet number one should choose regular grids.

In Figure 6.11 the example of advection dominance is shown. Péclet number equals to $6.58 \cdot 10^4$ and solutions on all grids have the similar character. From the error in Figure 6.9 we can conclude that time-of-flight indicator achieves the best accuracy, when advection dominates and that corresponds to results in [16]. Hence in the case of convection-dominated problem we can choose coarse grid based on time-of-flight or velocity indicators.

The same experiment we simulate on the 25th SPE 10 layer, which has similar permeability field as the 1st layer. From Figure 6.12 we can conclude that the error has similar behaviour as for the problem on the 1st SPE 10 layer. Cartesian grid has better accuracy at low Péclet numbers and in the region of equivalent dominance of the conductive and convective terms in the energy equation. Solutions on flow-based grids have significant error for conductive-dominated problem and better accuracy in comparison with Cartesian grid for convective-dominated problem.

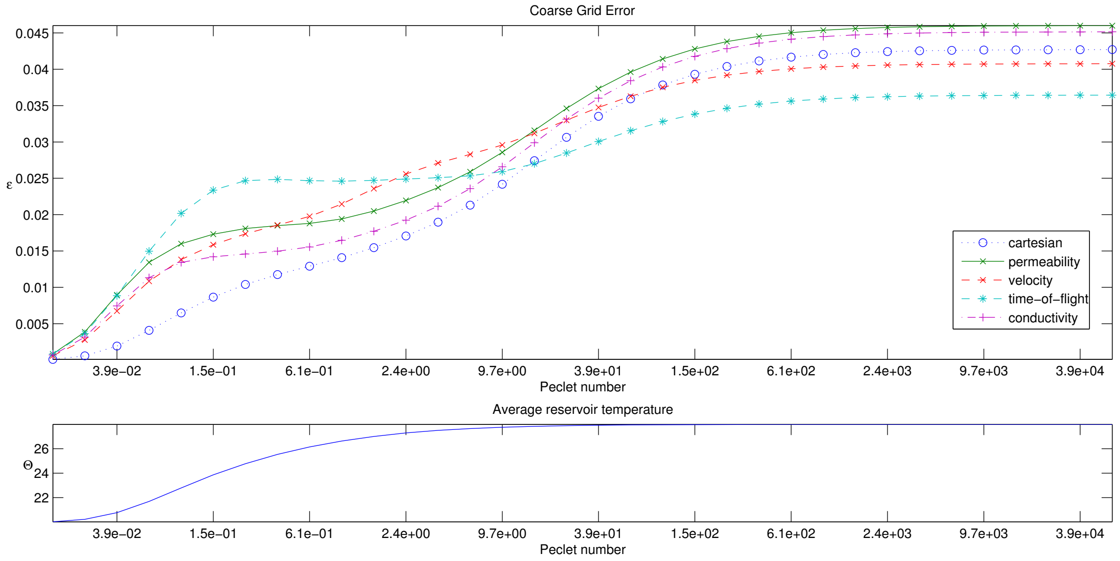


Figure 6.12: The error and average reservoir temperature for various Péclet numbers on the 25th SPE 10 layer heterogeneous medium.

6.3.3 Channeled heterogeneous medium

Here we consider a problem on the 44th layer of SPE 10 model. The layers from 37 to 85 in SPE 10 model have permeability and porosity fields that correspond to channeled reservoirs. In fluid transport upscaling Cartesian grid does not usually provide accurate solution for such permeability distribution because uniform coarse grids do not resolve channels with proper accuracy. The logarithmic scale of permeability field on the 44th layer is shown in Figure 6.14 and channeled structure is evident here. We chose this layer because of high permeability regions close to the right boundary and thus this example should reflect the heat transport in channels. Also according to [16] flow-based grids should give more accuracy in comparison to uniform grids in such flow channels.

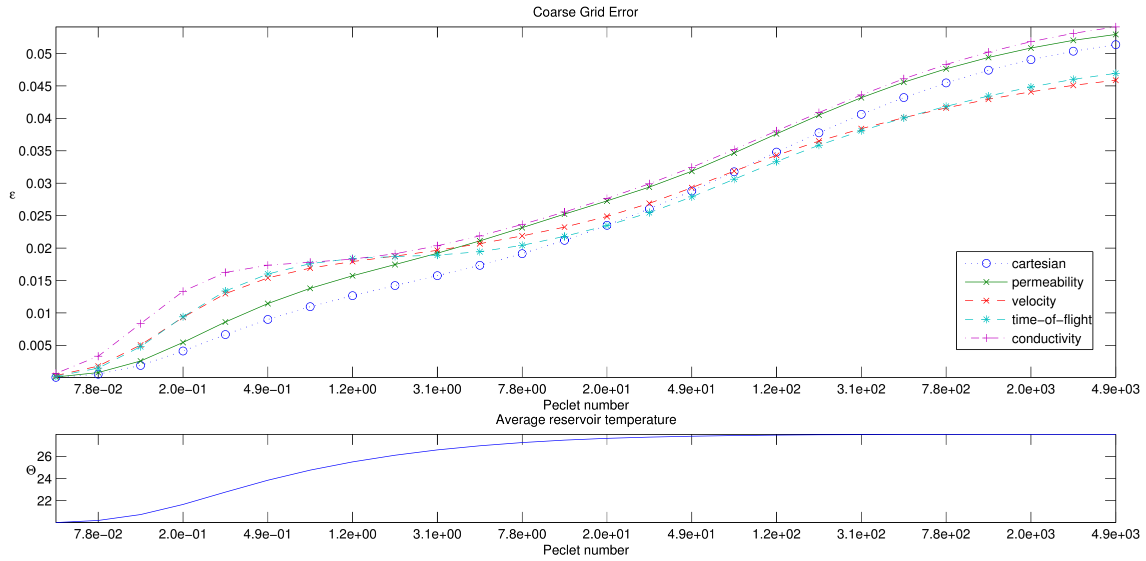


Figure 6.13: The error and average reservoir temperature for various Péclet numbers on the 44th SPE 10 layer heterogeneous medium.

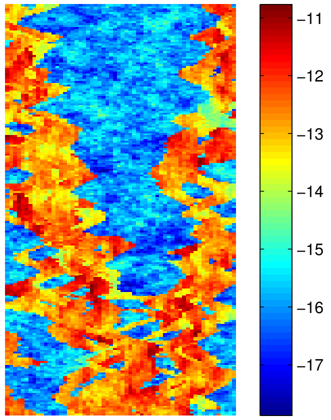


Figure 6.14: Common logarithm of permeability on the 44th SPE 10 layer.

We again solve the boundary flux problem that was set in section 6.2.2 and the heat and fluid parameters are defined in Table 6.3. We start our simulations with high thermal conductivity value and then reduce the dominance of the conductive term. For every simulation we solve the conservation of energy equation (3.7) implicitly on different grids as in the previous example. To build flow-based coarse grids we set the lower bound on cell volumes, N_L, equal to 15 and the upper bound on total amount of flow through each coarse grid block, N_U, equals to 30. The resulting coarse grids can be observed in Figures 6.15 and 6.16.

The error for different thermal conductivities is shown in Figure 6.13. We can see that all grids have small difference in accuracy for different Péclet numbers. In the region, where conduction dominates over convection, Cartesian grid has the best solution, permeability indicator based grid has the second-smallest error, but other flow-based grids have increased error. Similar behaviour was noticed in the previous experiments on layer 1 and 25. Again we can assume that such error is caused by irregularities in the

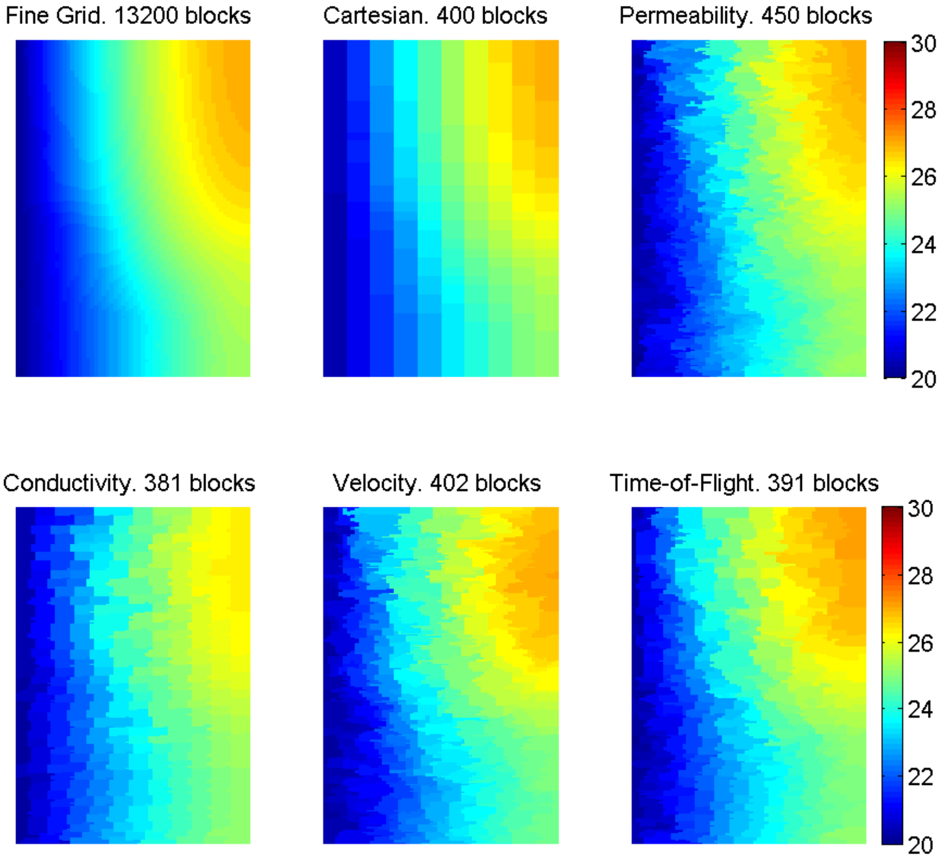


Figure 6.15: Simulations of the conductive-dominated energy equation on various grids on the 44th SPE 10 layer.

grid cells' shape and by the two-point flux approximation scheme. When Péclet numbers are high, coarse grids based on time-of-flight and velocity become more accurate. We can see now that the velocity indicator produces solution with less error in comparison to Figures 6.9 and 6.12. It seems that the solution based on conductivity indicator is the worst on this layer.

Solutions on the fine and coarse grids, when thermal conductivity equals to $3.9 \cdot 10^7$ W/(m· K), are shown in Figure 6.15. Péclet number in this case equals to 0.4936 and that corresponds to the situation, when the conductive term dominates in the energy equation. Obviously flow-based grid cells do not follow the contour of the temperature front. Such irregular shape of cells can lead to inconsistency of the two-point flux approximation scheme and we can not expect the numerical solution on such coarse grid to converge to the true solution.

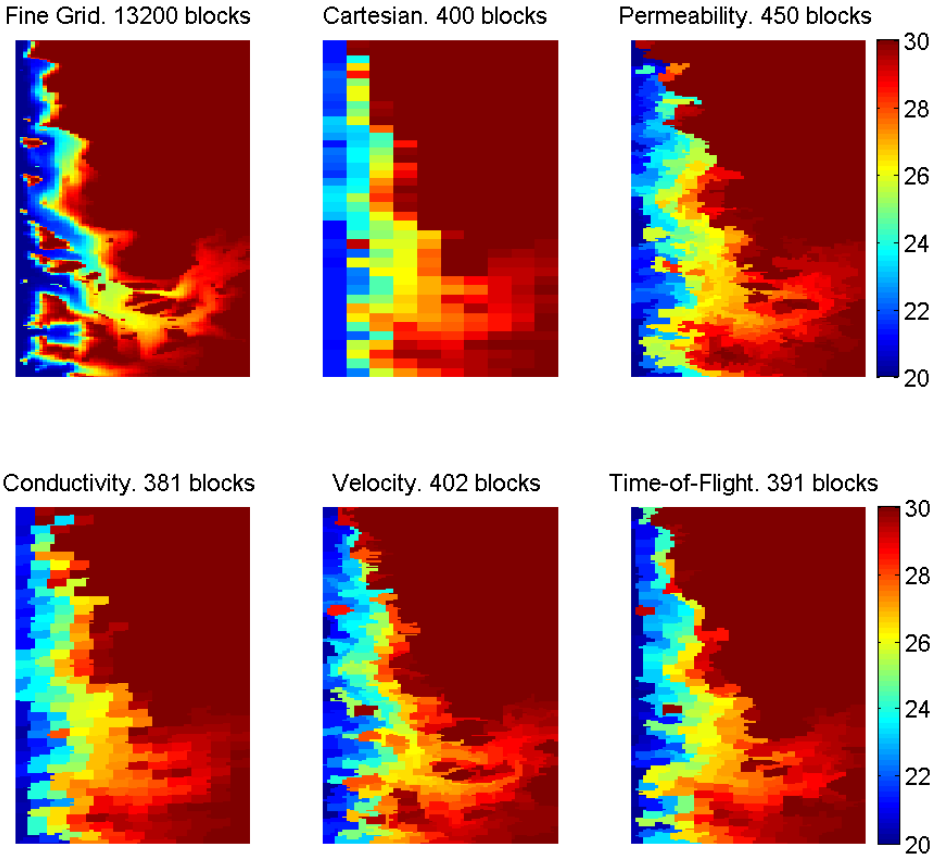


Figure 6.16: Simulations of the convective-dominated energy equation on various grids on the 44th SPE 10 layer.

Advection-dominated problem is shown in Figure 6.15, when the thermal conductivity equals to $3.9 \cdot 10^3$ W/(m· K). In this case Péclet number equals to $4.9 \cdot 10^3$ and obviously flow-based coarse grid cells follow the channeled contour. Cartesian grid and the coarse grid based on conductivity have diffused solutions and we can see that many flow details are lost. On the contrary flow-based coarse grids that were generated from permeability, velocity and time-of-flight better match to the fine grid solution. However, in this case permeability-based coarse grid is less accurate and it is more reasonable to use velocity or time-of-flight indicators. Hence we can conclude that in case of channeled reservoir and dominance of advection flow-based coarse grids based on the considered indicators produce definitely better solutions than uniform coarse grids.

Chapter 7

Conclusion

In this work we have investigated two different numerical solvers for heat transfer simulation in geothermal reservoirs. To solve the conservation of energy equation we used two schemes: Euler's forward method and Euler's backward method for temporal discretization and for spatial discretization the upwind and the two-point flux approximation were used. We applied these solvers to the problem with a heat flow through reservoir boundaries and compared obtained solutions, convergence and computational time for both methods. It was affirmed that the implicit scheme does not have any requirements on convergence in contrast to the explicit scheme and both methods have very similar solutions. However we found out that the numerical diffusion effect appears in Euler's backward method and to avoid such type of numerical error the time step should be reduced. Nevertheless, in many cases Euler's forward scheme requires too many time steps and it is feasible to use Euler's backward scheme.

We also upscaled and solved the convective-conductive energy equation for both homogeneous and heterogeneous media on various grids that are based on uniform and flow-based partitions. It is known from [16] that flow-based upgridding usually gives more accurate solution in fluid transport simulations. Flow-based coarse grids are very effective for convective processes in highly heterogeneous reservoirs and thus this coarsening strategy was chosen for our simulations. Our results show that different grids produce different errors and it is always hard to predict, which grid will give best accuracy for a particular kind of problem. Especially for the problem with boundary heat flux it seems reasonable to use different grids according to different Péclet numbers. According to our results flow-based grids may have better accuracy in comparison with the Cartesian grid in case of advection dominance. In our simulations time-of-flight indicator is usually one with the least error, while the solution obtained by velocity-based grid tends to have the best accuracy in the experiment in a channeled heterogeneous medium. Nevertheless, sometimes it can be better to use simple Cartesian grid even if advection dominates.

Solutions at low Péclet numbers, when advection has minimal influence in the equation, are also worthy of note. From our results we can conclude that the Cartesian grid has the smallest error both for homogeneous and heterogeneous media in this case, whereas flow-based grids produce significant error. It was also found out that the Cartesian grid is more effective when convection and conduction have approximately equal influence on the solution. Big accuracy error that produced by flow-based grids for a conduction-dominated problem can be caused by a coarse grid cells' shape that was generated by transport or flow oriented indicator. Such shape does not appropriately fit for the diffusive heat transfer. Also we suppose that such inadequate accuracy may arise from inconsistency of the two-point flux approximation discretization of the conductive term. This method was chosen for our solvers due to its wide utilization in reservoir simulation. It is a well-known scheme, that is simple, robust and computationally efficient. However, we must take into account that this method depends on a grid structure and if the grid is not K-orthogonal, then the two-point flux discretization leads to inconsistency. To avoid such error we advise to try some other schemes, that can handle a grid's structure in more accurate way, and hopefully our results and investigations will help in further heat transfer studying.

List of Symbols

Symbol	**Explanation**	**Units**
$\Gamma, \partial\Omega$	boundary of domain	
Γ_D	Dirichlet boundary	
Γ_N	Neumann boundary	
$\mathit{\Theta}$	temperature	K or ℃
Ω	domain	
$\mathit{\Psi}$	volumetric heat flux	W
α	half-transmissibility	$m/(s \cdot Pa)$
β	thermal expansion coefficient	K^{-1}
ϵ	global discretization error	
κ	hydraulic conductivity	m/s
μ	viscosity	$Pa \cdot s$
ρ	density	kg/m^3
τ	time-of-flight or	s
	tortuosity	
ϕ	effective porosity	
$\vec{\psi}$	heat flux	W/m^2
A	cross-sectional area	m^2
B	formation volume factor	
C	heat capacity	J/K
J	well injectivity index	$m/(s \cdot Pa)$
L	length	m
$\boldsymbol{K}$	permeability	m^2 or D
Q	volumetric flow rate	m^3/s
δQ	heat energy	J
Pe	Péclet number	
Pr	Prandtl number	
Re	Reynolds number	

Symbol	**Explanation**	**Units**
R	truncation error	
S	entropy	J/K
T	transmissibility	$m/(s \cdot Pa)$
U	energy	J
V	volume	m^3
$\vec{V}$	intrinsic fluid velocity	m/s
W	work	J
c	specific heat capacity	$J/(kg \cdot K)$
c_p	specific heat capacity at the constant pressure	$J/(kg \cdot K)$
c_V	specific heat capacity at the constant volume	$J/(kg \cdot K)$
c_f	fluid compressibility	Pa^{-1}
c_r	rock compressibility	Pa^{-1}
e	energy density	J/m^3
f	flux	m^3/s
$\boldsymbol{k}$	thermal conductivity	$W/(m \cdot K)$
g	acceleration due to gravity	m/s^2
h	hydraulic head or	m
	height	m
l	length	m
m	mass	kg
$\vec{n}$	normal	
q	volumetric flow rate	m^3/s
p	pressure	Pa or $kg/(m \cdot s^2)$
r	radius	m
$\vec{v}$	velocity,flow rate	m/s
z	vertical coordinate, depth	m

List of Figures

1.1 Enhanced geothermal system. 4

2.1 Porous Medium. 5
2.2 Volume Ω with boundary $\partial\Omega$ and outward normal $\vec{n}$. 9

3.1 The conservation of energy law. 19

4.1 Two-point flux discretization for grid cells Ω_i and Ω_k. 25

5.1 Volume averaging of porosity. 34
5.2 Coarse grid block. 35
5.3 The resulting coarse grid for different indicators. 37
5.4 Four steps of the flow-based coarsening algorithm. 38

6.1 Pressure [Pa] and time-of-flight [s] for a homogeneous medium. . . 40
6.2 Common logarithm of permeability, velocity and time-of-flight for the boundary flux problem on the 1st SPE 10 layer. 41
6.3 Error of the implicit scheme with various numbers of time steps for the boundary flux problem and plot of the function $f(x) = 1/x$ at logarithmic scales of axes. 44
6.4 Temperature distribution in reservoir for different final time. Plots (a), (c), (e) correspond to Péclet number of 2 and plots (b), (d), (f) correspond to low Péclet number of 0.02. 45
6.5 Temperature distribution for two problems with different velocities. 47
6.6 Numerical diffusion effect in the implicit scheme with different number of time steps in comparison to the explicit scheme. 48
6.7 Simulations of the energy equation with two different Péclet numbers of 200 and 0.02 on various grids in a homogeneous medium. . 51
6.8 The error and average reservoir temperature for various Péclet numbers on a homogeneous medium. 52
6.9 The error and average reservoir temperature for various Péclet numbers on the 1st SPE 10 layer heterogeneous medium. 53

6.10 Simulations of the conductive-dominated energy equation on various grids on the 1st SPE 10 layer. 54
6.11 Simulations of the convective-dominated energy equation on various grids on the 1st SPE 10 layer. 55
6.12 The error and average reservoir temperature for various Péclet numbers on the 25th SPE 10 layer heterogeneous medium. 56
6.13 The error and average reservoir temperature for various Péclet numbers on the 44th SPE 10 layer heterogeneous medium. 57
6.14 Common logarithm of permeability on the 44th SPE 10 layer. . . 57
6.15 Simulations of the conductive-dominated energy equation on various grids on the 44th SPE 10 layer. 58
6.16 Simulations of the convective-dominated energy equation on various grids on the 44th SPE 10 layer. 59

List of Tables

3.1 Thermal conductivities in W/(m · K) [36] 16
3.2 Specific heat capacities in J/(kg · K) [34], [32] 17

6.1 Euler's forward method's computational time. 42
6.2 Euler's backward method's computational time and error 43
6.3 Heat parameters in experiments. 46
6.4 Euler's forward method's computational time. 46
6.5 Euler's backward method's computational time and error. 47
6.6 Euler's forward method's computational time 48
6.7 Euler's backward method's computational time and error for problem with high Péclet number. 49
6.8 Euler's backward method's computational time and error for problem with low Péclet number 49

Bibliography

[1] Aarnes, J. E., Hauge, V. L., and Efendiev, Y. (2007). Coarsening of three-dimensional structured and unstructured grids for subsurface flow. *Advances in Water Resources*, 30(11):2177–2193.

[2] Aziz, K., Durlofsky, L., and Tchelepi, H. (2004). Notes on reservoir simulation.

[3] Aziz, K. and Settari, A. (1979). *Petroleum reservoir simulation.* Chapman & Hall.

[4] Bear, J. (2013). *Dynamics of fluids in porous media.* Courier Corporation.

[5] Bennett, G. D., Reilly, T. E., and Hill, M. C. (1990). *Technical training notes in ground-water hydrology: radial flow to a well.* Department of the Interior, US Geological Survey.

[6] Carman, P. C. (1956). *Flow of gases through porous media.* Academic press.

[7] Deutsch, C. et al. (1989). Calculating effective absolute permeability in sandstone/shale sequences. *SPE Formation Evaluation*, 4(03):343–348.

[8] Durlofsky, L. J. (2003). Upscaling of geocellular models for reservoir flow simulation: a review of recent progress. In *7th International Forum on Reservoir Simulation Bühl/Baden-Baden, Germany*, pages 23–27. Citeseer.

[9] Durlofsky, L. J. (2005). Upscaling and gridding of fine scale geological models for flow simulation. In *8th International Forum on Reservoir Simulation Iles Borromees, Stresa, Italy*, volume 2024.

[10] Eikemo, B., Berre, I., Dahle, H., Lie, K., and Natvig, J. (2006). A discontinouos galerkin method for computing time-of-flight in discrete-fracture models. In *Proceedings of the XVI International Conference on Computational Methods in Water Resources, Copenhagen, Denmark*, pages 18–22.

[11] Evans, L. (2010). *Partial Differential Equations.* Graduate studies in mathematics. American Mathematical Society.

[12] Fridleifsson, I. B., Bertani, R., Huenges, E., Lund, J. W., Ragnarsson, A., Rybach, L., et al. (2008). The possible role and contribution of geothermal energy to the mitigation of climate change. In *IPCC scoping meeting on renewable energy sources, proceedings, Luebeck, Germany*, volume 20, pages 59–80. Citeseer.

[13] Gautschi, W. (2011). *Numerical analysis*. Springer Science & Business Media.

[14] GEA (2015). The international geothermal market at a glance.

[15] Hantush, M. S. (1964). Hydraulics of wells. *Advances in hydroscience*, 1:281–432.

[16] Hauge, V. L. (2010). Multiscale methods and flow-based gridding for flow and transport in porous media.

[17] Hauge, V. L., Lie, K.-A., and Natvig, J. R. (2011). Grid coarsening based on amalgamation for multi-fidelity transport solvers.

[18] Hauge, V. L., Lie, K.-A., and Natvig, J. R. (2012). Flow-based coarsening for multiscale simulation of transport in porous media. *Computational Geosciences*, 16(2):391–408.

[19] Lay, D. C. (2003). Linear algebra and its applications. *Addison-Wesley Publishing Company*, 3:124.

[20] Lie, K.-A. (2014). An introduction to reservoir simulation using matlab. *SINTEF ICT*.

[21] Lie, K.-A., Krogstad, S., Ligaarden, I. S., Natvig, J. R., Nilsen, H. M., and Skaflestad, B. (2012). Open-source matlab implementation of consistent discretisations on complex grids. *Computational Geosciences*, 16(2):297–322.

[22] Lienhard, J. H. (2013). *A heat transfer textbook*. Courier Corporation.

[23] Mims, C. (2009). Can geothermal power compete with coal on price. *Scientific American*, 55.

[24] Nedreli, A. C. (2014). Transport upscaling in porous media.

[25] Nield, D. and Bejan, A. (1999). *Convection in Porous Media*. Springer, New York.

[26] Nordbotten, J. and Celia, M. (2011). *Geological Storage of CO2: Modeling Approaches for Large-Scale Simulation*. Wiley.

[27] Patankar, S. (1980). *Numerical heat transfer and fluid flow.* CRC Press.

[28] Pollack, H. N., Hurter, S. J., and Johnson, J. R. (1993). Heat flow from the earth's interior: analysis of the global data set. *Reviews of Geophysics*, 31(3):267–280.

[29] Sandve, T. H. (2013). Multiscale simulation of flow and heat transport in fractured geothermal reservoirs.

[30] Sawin, J., Sverrisson, F., and Rickerson, W. (2015). Renewables 2015 Global Status Report, REN21.

[31] SINTEF. Matlab Reservoir Simulation Toolbox. `http://www.sintef.no/Projectweb/MRST/`. Accessed: November 2015.

[32] Solids - Specific Heat. `http://www.engineeringtoolbox.com/specific-heat-solids-d_154.html`. Accessed: November 2015.

[33] SPE Comparative Solution Project. Description of model 2. `http://www.spe.org/web/csp/datasets/set02.htm`. Accessed: November 2015.

[34] Specific Heat of some common Substances. `http://www.engineeringtoolbox.com/specific-heat-capacity-d_391.html`. Accessed: November 2015.

[35] Tester, J. W., Anderson, B., Batchelor, A., Blackwell, D., DiPippo, R., Drake, E., Garnish, J., Livesay, B., Moore, M. C., Nichols, K., et al. (2006). The future of geothermal energy: Impact of enhanced geothermal systems (egs) on the united states in the 21st century. *Massachusetts Institute of Technology*, 209.

[36] Thermal conductivity of some common Materials and Gases. `http://www.engineeringtoolbox.com/thermal-conductivity-d_429.html`. Accessed: November 2015.

[37] Welty, J. R., Wicks, C. E., Rorrer, G., and Wilson, R. E. (2009). *Fundamentals of momentum, heat, and mass transfer.* John Wiley & Sons.