4

Mathematical Models for Single-Phase Flow

If you have read the chapters of the book in chronological order, you have already encountered the equations modeling flow of a single, incompressible fluid through a porous media twice: first in Section 1.4, which showed how vertical equilibrium inside can be computed by use of MRST, and then in Section 2.4.2, in which we discussed the concept of rock permeability. In this chapter, we review mathematical models for single-phase flow in more detail, including auxiliary boundary and initial conditions, subscale models for injection and production wells, and steady-state advection equations. We then introduce basic spatial discretizations for the resulting equations and show how these discretizations can be formulated as abstract discrete differentiation operators. A more thorough discussion of how these discretizations can be turned into efficient solvers is deferred to Chapters 5 and 6, which focus on the incompressible case, and Chapter 7, which discusses compressible flow and how the discrete differentiation operators can be combined with automatic differentiation for rapid prototyping of new solvers.

4.1 Fundamental Concept: Darcy's Law

Macroscale modeling of single-phase flow in porous media started with the work of Henry Darcy, a French hydraulic engineer, who in the middle of the nineteenth century was engaged to enlarge and modernize the waterworks of the city of Dijon. To understand the physics of flow through the sand filters that were used to clean the water supply, Darcy designed a vertical experimental tank filled with sand, in which water was injected at the top and allowed to flow out at the bottom; Figure 4.1 shows a conceptual illustration. Once the sand pack is filled with water, and the inflow and outflow rates are equal, the hydraulic head at the inlet and at the outlet can be measured using mercury-filled manometers. The hydraulic head is given as $h = E/mg = z - p/\rho g$, relative to a fixed datum. As water flows through the porous medium, it will experience a loss of energy. In a series of experiments, Darcy measured the water volumetric flow rate out of the tank and compared this rate with the loss of hydrostatic head from top to bottom of the column. From the experiments, he established that for the same sand pack, the discharge (flow rate) Q [m^3/s] is proportional

https://doi.org/10.1017/9781108591416.007 Published online by Cambridge University Press

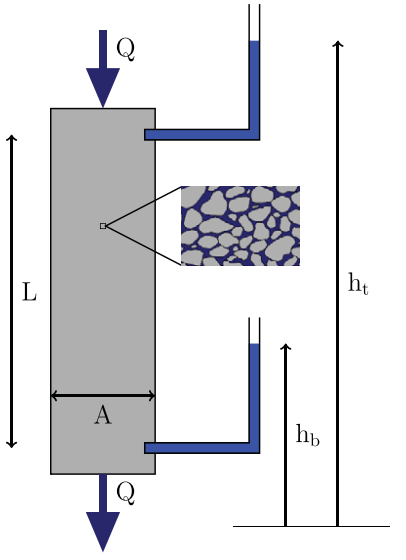


Figure 4.1 Conceptual illustration of Darcy's experiment.

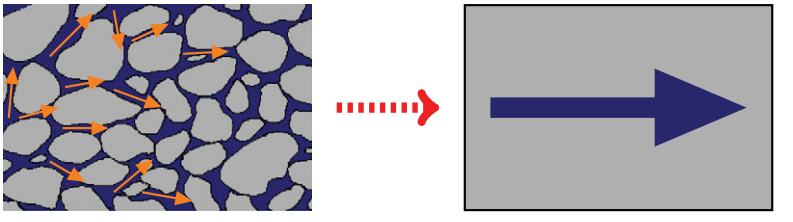


Figure 4.2 The macroscopic Darcy velocity represents an average of microscopic fluid fluxes.

to the cross-sectional area A [m^2] and the difference in hydraulic head (height of the water) $h_t - h_b$ [m], and inversely proportional to the flow length of the tank L [m]. Altogether, this can be summarized as

$$\frac{Q}{A} = \kappa \frac{h_t - h_b}{L}. \tag{4.1}$$

The result was presented in 1856 as an appendix to [77] entitled "Determination of the laws of flow of water through sand" and is the direct predecessor to what we today call Darcy's law. In (4.1), κ [m/s] denotes the hydraulic conductivity, which is a function both of the medium and the fluid flowing through it. It follows from a dimensional analysis that $\kappa = \rho g K/\mu$, where g [m/s^2] is the gravitational acceleration, μ [kg/ms] is the dynamic viscosity, and K [m^2] is the intrinsic permeability of a given sand pack.

The specific discharge $v = Q/A$, or Darcy flux, through the sand pack represents the volume of fluid flowing per total area of the porous medium and has dimensions [m/s].

https://doi.org/10.1017/9781108591416.007 Published online by Cambridge University Press

Somewhat misleading, v is often referred to as a velocity. However, since only a fraction of the cross-sectional area is available for flow (the majority of the area is blocked by sand grains), v is not a velocity in the microscopic sense. Instead, v is the apparent macroscopic velocity obtained by averaging the microscopic fluxes inside representative elementary volumes (REVs), which were discussed in Section 2.3.2. The macroscopic fluid velocity, defined as volume of fluid flowing per area occupied by fluid, is therefore given by v/ϕ, where ϕ is the porosity associated with the REV.

Henceforth, we will, with a slight abuse of notation, refer to the specific discharge as the *Darcy velocity*. In modern differential notation, first proposed by Nutting [239] and Wyckoff et al. [320], Darcy's law for a single-phase fluid reads,

$$\vec{v} = -\frac{\mathbf{K}}{\mu}(\nabla p - g\rho\nabla z), \tag{4.2}$$

where p is the fluid pressure and z is the vertical coordinate. The equation expresses conservation of momentum and was derived from the Navier–Stokes equations by averaging and neglecting inertial and viscous effects by Hubbert [137] and later from Stokes flow by Whitaker [315]. The observant reader will notice that Darcy's law (4.2) is analogous to Fourier's law (1822) for heat conduction, Ohm's law (1827) in the field of electrical networks, or Fick's law (1855) for fluid concentrations in diffusion theory, except that for Darcy there are two driving forces, pressure and gravity. Notice also that Darcy's law assumes a reversible fluid process, which is a special case of the more general physical laws of irreversible processes that were first described by Onsager (1931).

4.2 General Flow Equations for Single-Phase Flow

To derive a mathematical model for single-phase flow on the macroscopic scale, we first make a continuum assumption based on the existence of REVs as discussed in the previous section and then look at a control volume as shown in Figure 4.3. From the fundamental

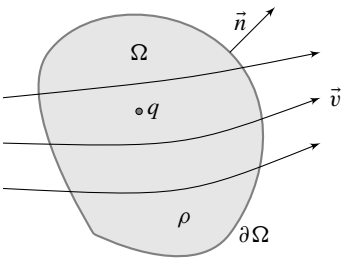


Figure 4.3 Illustration of a control volume Ω on which one can apply the principle of conservation to derive macroscopic continuity equations.

https://doi.org/10.1017/9781108591416.007 Published online by Cambridge University Press

law of mass conservation, we know that the accumulation of mass inside this volume must equal the net flux over the boundaries,

$$\frac{\partial}{\partial t}\int_{\Omega}\phi\rho\, d\vec{x} + \int_{\partial\Omega}\rho\vec{v}\cdot\vec{n}\, ds = \int_{\Omega}\rho q\, d\vec{x}, \tag{4.3}$$

where ρ is the density of the fluid, ϕ is the rock porosity, $\vec{v}$ is the macroscopic Darcy velocity, $\vec{n}$ denotes the normal at the boundary $\partial\Omega$ of the computational domain Ω, and q denotes fluid sources and sinks, i.e., outflow and inflow of fluids per volume at certain locations. Applying Gauss's theorem, this conservation law can be written in the alternative integral form

$$\int_{\Omega}\left[\frac{\partial}{\partial t}\phi\rho + \nabla\cdot(\rho\vec{v})\right] d\vec{x} = \int_{\Omega}\rho q\, d\vec{x}. \tag{4.4}$$

This equation is valid for any volume Ω, and in particular volumes that are infinitesimally small. Hence, it follows that the macroscopic behavior of the single-phase fluid must satisfy the continuity equation

$$\frac{\partial(\phi\rho)}{\partial t} + \nabla\cdot(\rho\vec{v}) = \rho q. \tag{4.5}$$

Equation (4.5) contains more unknowns than equations, and to derive a closed mathematical model, we need to introduce what are commonly referred to as *constitutive equations* that give the relationship between different states of the system (pressure, volume, temperature, etc.) at given physical conditions. Darcy's law, discussed in the previous section, is an example of a constitutive relation that has been derived to provide a phenomenological relationship between the macroscale $\vec{v}$ and the fluid pressure p. In Section 2.4.1 we introduced the rock compressibility $c_r = d\ln(\phi)/dp$, which describes the relationship between the porosity ϕ and the pressure p. In a similar way, we can introduce the fluid compressibility to relate the density ρ to the fluid pressure p.

A change in density will generally cause a change in both pressure p and temperature T. The usual way of describing these changes in thermodynamics is to consider the change of volume V for a fixed number of particles,

$$\frac{dV}{V} = \frac{1}{V}\left(\frac{\partial V}{\partial p}\right)_T dp + \frac{1}{V}\left(\frac{\partial V}{\partial T}\right)_p dT, \tag{4.6}$$

where the subscripts T and p indicate that the change takes place under constant temperature and pressure, respectively. Since ρV is constant for a fixed number of particles, $V d\rho = -\rho dV$, and (4.6) can be written in the equivalent form

$$\frac{d\rho}{\rho} = \frac{1}{\rho}\left(\frac{\partial\rho}{\partial p}\right)_T dp + \frac{1}{\rho}\left(\frac{\partial\rho}{\partial T}\right)_p dT = c_f dp + \alpha_f dT. \tag{4.7}$$

Here, c_f denotes the *isothermal compressibility* and α_f the *thermal expansion coefficient*. In many subsurface systems, the density changes slowly so that heat conduction keeps the temperature constant, in which case (4.7) simplifies to

https://doi.org/10.1017/9781108591416.007 Published online by Cambridge University Press

$$c_f = \frac{1}{\rho}\frac{d\rho}{dp} = \frac{d\ln(\rho)}{dp}. \tag{4.8}$$

The factor c_f, which we henceforth will refer to as the *fluid compressibility*, is nonnegative and will generally depend on both pressure and temperature, i.e., $c_f = c_f(p,T)$.

Introducing Darcy's law and fluid and rock compressibilities in (4.5), we obtain the parabolic equation for the fluid pressure

$$c_t\phi\rho\frac{\partial p}{\partial t} - \nabla\cdot\left[\frac{\rho\mathbf{K}}{\mu}(\nabla p - g\rho\nabla z)\right] = \rho q, \tag{4.9}$$

where $c_t = c_r + c_f$ denotes the *total compressibility*. Notice that this equation is generally nonlinear since both ρ and c_t may depend on p. In the following, we will look briefly at several special cases in which the governing single-phase equation becomes a linear equation for the primary unknown; more extensive discussions can be found in standard textbooks like [246, chap. 1], [69, chap. 2]. For completeness, we also briefly review the concept of an equation of state.

Incompressible flow. In the special case of an incompressible rock and fluid – that is, ρ and ϕ are independent of p so that $c_t = 0$ – (4.9) simplifies to an elliptic equation with variable coefficients,

$$-\nabla\cdot\left[\frac{\mathbf{K}}{\mu}\nabla(p - g\rho z)\right] = q. \tag{4.10}$$

If we introduce the fluid potential, $\Phi = p - g\rho z$, (4.10) can be recognized as a variable-coefficient Poisson's equation $-\nabla\cdot\mathbf{K}\nabla\Phi = q$ or as the Laplace equation $\nabla\cdot\mathbf{K}\nabla\Phi = 0$ if there are no volumetric fluid sources or sinks. In the next section, we will discuss in detail how to discretize the second-order spatial Laplace operator $\mathcal{L} = \nabla\cdot\mathbf{K}\nabla$, which is a key technological component that will enter almost any software for simulating flow in porous rock formations.

Constant compressibility. If the fluid compressibility is constant and independent of pressure, (4.8) can be integrated from a known density ρ_0 at a pressure datum p_0 to give the equation

$$\rho(p) = \rho_0 e^{c_f(p-p_0)}, \tag{4.11}$$

which applies well to most liquids that do not contain large quantities of dissolved gas. To develop the differential equation, we first assume that the porosity and the fluid viscosity do not depend on pressure. Going back to the definition of fluid compressibility (4.8), it also follows from this equation that $\nabla p = (c_f\rho)^{-1}\nabla\rho$, which we can use to eliminate ∇p from Darcy's law (4.2). Inserting the result into (4.5) gives us the continuity equation

$$\frac{\partial\rho}{\partial t} - \frac{1}{\mu\phi c_f}\nabla\cdot\left(\mathbf{K}\nabla\rho - c_f g\rho^2\mathbf{K}\nabla z\right) = \rho q. \tag{4.12}$$

https://doi.org/10.1017/9781108591416.007 Published online by Cambridge University Press

In the absence of gravity forces and source terms, this is a linear equation for the fluid density that is similar to the classical heat equation with variable coefficients,

$$\frac{\partial \rho}{\partial t} = \frac{1}{\mu \phi c_f} \nabla \cdot (\mathbf{K} \nabla \rho) \,. \tag{4.13}$$

Slightly compressible flow. If fluid compressibility is small, it is sufficient to use a linear relationship to evaluate the exponential form as

$$\rho = \rho_0 \left[1 + c_f (p - p_0) \right]. \tag{4.14}$$

We further assume that ϕ has a similar functional dependence and that μ is constant. For simplicity, we also assume that g and q are both zero. Then, we can simplify (4.9) as follows:

$$\left[(c_r + c_f) \phi \rho \right] \frac{\partial p}{\partial t} = \frac{c_f \rho}{\mu} \nabla p \cdot \mathbf{K} \nabla p + \frac{\rho}{\mu} \nabla \cdot (\mathbf{K} \nabla p).$$

If c_f is sufficiently small, in the sense that $c_f \nabla p \cdot \mathbf{K} \nabla p \ll \nabla \cdot (\mathbf{K} \nabla p)$, we can neglect the first term on the right-hand side to derive a linear equation similar to (4.13) for the fluid pressure (introducing total compressibility $c = c_r + c_f$)

$$\frac{\partial p}{\partial t} = \frac{1}{\mu \phi c} \nabla \cdot (\mathbf{K} \nabla p) \,. \tag{4.15}$$

Ideal gas. If the fluid is a gas, compressibility can be derived from the gas law, which for an ideal gas can be written in two alternative forms,

$$pV = nRT, \qquad \rho = p(\gamma - 1)e. \tag{4.16}$$

In the first form, T is temperature, V is volume, $R = 8.314 \text{ J K}^{-1}\text{mol}^{-1}$ is the gas constant, and $n = m/M$ is the amount of substance of the gas in moles, where m is the mass and M is the molecular weight. In the second form, γ is the adiabatic constant, i.e., the ratio of specific heat at constant pressure and constant volume, and e is the specific internal energy (internal energy per unit mass). In either case, it follows from (4.8) that $c_f = 1/p$.

Gravity effects are negligible for gases and for brevity we once again we assume that ϕ is a function of $\vec{x}$ only. Inserting (4.16) into (4.9) then gives

$$\frac{\partial (\rho \phi)}{\partial t} = \phi(\gamma - 1)e \frac{\partial p}{\partial t} = \frac{1}{\mu} \nabla \cdot (\rho \mathbf{K} \nabla p) = \frac{(\gamma - 1)e}{\mu} \nabla \cdot (p \mathbf{K} \nabla p) \,,$$

from which it follows that

$$\phi \mu \frac{\partial p}{\partial t} = \nabla \cdot (p \mathbf{K} \nabla p) \qquad \Leftrightarrow \qquad \frac{\phi \mu}{p} \frac{\partial p^2}{\partial t} = \nabla \cdot \left(\mathbf{K} \nabla p^2 \right). \tag{4.17}$$

Equation of state. Equation (4.11), (4.14), and (4.16) are all examples of what is commonly referred to as equations of state, which provide constitutive relationships between mass, pressures, temperature, and volumes at thermodynamic equilibrium. Another popular form

https://doi.org/10.1017/9781108591416.007 Published online by Cambridge University Press

of these equations are the so-called cubic equations of state, which can be written as cubic functions of the molar volume $V_m = V/n = M/\rho$ involving constants that depend on pressure p_c, temperature T_c, and the molar volume V_c at the critical point, i.e., the point at which $\left(\frac{\partial p}{\partial V}\right)_T = \left(\frac{\partial^2 p}{\partial V^2}\right)_T \equiv 0$. A few particular examples include the Redlich–Kwong equation of state

$$p = \frac{RT}{V_m - b} - \frac{a}{\sqrt{T}\, V_m(V_m + b)},$$
$$a = \frac{0.42748 R^2 T_c^{5/2}}{p_c}, \quad b = \frac{0.08662 RT_c}{p_c}, \tag{4.18}$$

the modified version called Redlich–Kwong–Soave

$$p = \frac{RT}{V_m - b} - \frac{a\alpha}{\sqrt{T}\, V_m(V_m + b)},$$
$$a = \frac{0.427 R^2 T_c^2}{p_c}, \quad b = \frac{0.08664 RT_c}{p_c}, \tag{4.19}$$
$$\alpha = \left[1 + \left(0.48508 + 1.55171\omega - 0.15613\omega^2\right)\left(1 - \sqrt{T/T_c}\right)\right]^2,$$

as well as the Peng–Robinson equation of state,

$$p = \frac{RT}{V_m - b} - \frac{a\alpha}{V_m^2 + 2bV_m - b^2)},$$
$$a = \frac{0.4527235 R^2 T_c^2}{p_c}, \quad b = \frac{0.077796 RT_c}{p_c}, \tag{4.20}$$
$$\alpha = \left[1 + \left(0.37464 + 1.54226\omega - 0.26992\omega^2\right)\left(1 - \sqrt{T/T_c}\right)\right]^2.$$

Here, ω denotes the acentric factor of the species, which is a measure of the centricity (deviation from spherical form) of the molecules in the fluid. The Peng–Robinson model is much better at predicting liquid densities than the Redlich–Kwong–Soave model, which was developed to fit pressure data of hydrocarbon vapor phases. If we introduce an attractive factor A, a repulsive factor B, and the gas deviation factor Z (to be discussed in Section 11.4.3 on page 373)

$$A = \frac{a\alpha p}{(RT)^2}, \qquad B = \frac{bp}{RT}, \qquad Z = \frac{pV}{RT},$$

the Redlich–Kwong–Soave equation (4.19) and the Peng–Robinson equation (4.20) can be written in alternative polynomial forms,

$$0 = Z^3 - Z^2 + Z(A - B - B^2) - AB, \tag{4.21}$$
$$0 = Z^3 - (1 - B)Z^2 + (A - 2B - 3B^2)Z - (AB - B^2 - B^3), \tag{4.22}$$

which illustrates why they are called cubic equations of state.

https://doi.org/10.1017/9781108591416.007 Published online by Cambridge University Press

4.3 Auxiliary Conditions and Equations

The governing equations for single-phase flow discussed in the previous section are all parabolic, except for the incompressible case in which the governing equation is elliptic. For the solution to be well-posed[1] inside a finite domain for any of the equations, one needs to supply conditions that determine the behavior on the external boundary. For the parabolic equations describing unsteady flow, we also need to impose an initial condition that determines the initial state of the fluid system. This section discusses these conditions in more detail. We also discuss models for representing flow in and out of the reservoir rock through wellbores. Flow near wellbores typically takes place on a length scale much smaller than the length scales of the global flow inside the reservoir, and it is customary to model it using special analytical models. Finally, we also discuss a set of auxiliary equations for describing the movement of fluid elements and/or neutral particles that follow the single-phase flow without affecting it.

4.3.1 Boundary and Initial Conditions

In reservoir simulation one is often interested in describing closed flow systems that have no fluid flow across its external boundaries. This is a natural assumption when studying full reservoirs that have trapped and contained petroleum fluids for million of years. Mathematically, no-flow conditions across external boundaries are modeled by specifying *homogeneous Neumann conditions*,

$$\vec{v} \cdot \vec{n} = 0 \quad \text{for } \vec{x} \in \partial\Omega. \tag{4.23}$$

With no-flow boundary conditions, any pressure solution of (4.10) is immaterial and only defined up to an additive constant, unless a datum value is prescribed at some internal point or along the boundary.

It is also common that parts of the reservoir may be in communication with a larger aquifer system that provides external pressure support, which can be modeled in terms of a *Dirichlet condition* of the form

$$p(\vec{x}) = p_a(\vec{x}, t) \quad \text{for } \vec{x} \in \Gamma_a \subset \partial\Omega. \tag{4.24}$$

The function p_a can, for instance, be given as a hydrostatic condition. Alternatively, parts of the boundary may have a certain prescribed influx, which can be modeled in terms of an *inhomogeneous Neumann condition*,

$$\vec{v} \cdot \vec{n} = u_a(\vec{x}, t) \quad \text{for } \vec{x} \in \Gamma_a \subset \partial\Omega. \tag{4.25}$$

Combinations of these conditions are used when studying parts of a reservoir (e.g., sector models). There are also cases, for instance when describing groundwater systems or CO_2

[1] A solution is well-posed if it exists, is unique, and depends continuously on the initial and boundary conditions.

https://doi.org/10.1017/9781108591416.007 Published online by Cambridge University Press

sequestration in saline aquifers, where (parts of) the boundaries are open, or the system contains a background flow. More information of how to set boundary conditions will be given in Section 5.1.4. In the compressible case in (4.9), we also need to specify an initial pressure distribution. Typically, this pressure distribution will be hydrostatic, as in the gravity column we discussed briefly in Section 1.4, and hence be given by the ordinary differential equation,

$$\frac{dp}{dz} = \rho g, \qquad p(z_0) = p_0. \tag{4.26}$$

4.3.2 Injection and Production Wells

In a typical reservoir simulation, the inflow and outflow in wells occur on a subgrid scale. In most discretized flow models, the pressure is modeled using a single pressure value inside each grid cell. The size of each grid cell must therefore be chosen so small that the pressure variation inside the cell can be approximated accurately in terms of its volumetric average. Far away from wells, spatial variations in pressure tend to be relatively slow, at least in certain directions, and one can therefore choose cell sizes in the order of tens or hundreds of meters, which is a reasonable size compared with the extent of the reservoir. Near the well, however, the pressure will have large variations over short distances, and to compute a good approximation of these pressure variations, one would need grid cells than are generally smaller than what is computationally tractable. As a result, one ends up with a setup similar to the illustration in Figure 4.4, where the radius of the well is typically between 1/100 and 1/1000 of the horizontal dimensions of the grid cell. The largest percentage of the pressure drop associated with a well occurs near the well and the pressure at the well radius will thus deviate significantly from the volumetric pressure average inside the cell. Special analytical models are therefore developed to represent the subgrid variations in the particular flow patterns near wells.

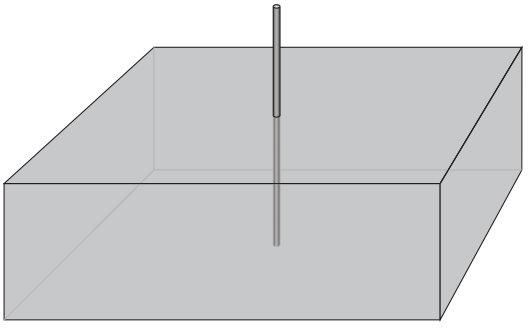


Figure 4.4 Illustration of a well inside a grid cell. The proportions are not fully to scale: whereas the diameter of a well varies from 5 to 40 inches, a grid block may extend from tens to hundreds of meters in the lateral direction and from a few decimeters to ten meters in the vertical direction.

https://doi.org/10.1017/9781108591416.007 Published online by Cambridge University Press

Inflow-Performance Relation

Normally, fluids are injected from a well at either constant *surface rate* or at constant *bottom-hole pressure*, which is also called wellbore flowing pressure and refers to the pressure at a certain datum point inside the wellbore. Similarly, fluids are produced at constant bottom-hole pressure or at constant surface liquid rate. The main purpose of a well model is then to accurately compute the pressure at the well radius when the injection or production rate is known, or to accurately compute the flow rate in or out of the reservoir when the pressure at well radius is known. The resulting relation between the bottom-hole pressure and surface flow rate is often called the *inflow-performance relation* or IPR.

The simplest and most widely used inflow–performance relation is the linear law

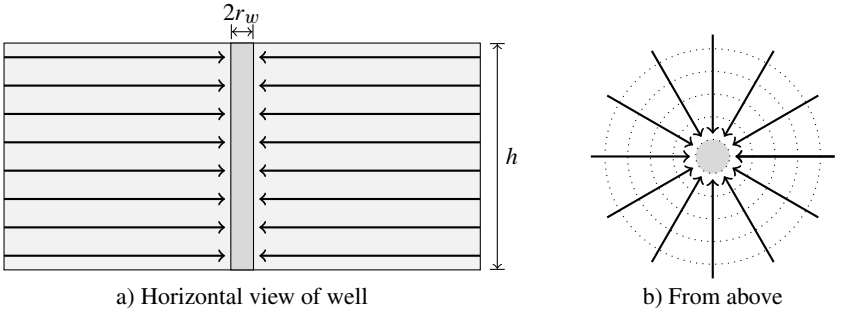
$$q_o = J(p_R - p_w), \tag{4.27}$$

which states that the flow rate is directly proportional to the pressure drawdown near the well; that is, flow rate is proportional to the difference between the average reservoir pressure p_R in the grid cell and the pressure p_w at the wellbore. The constant of proportionality J is called the *productivity index* (PI) for production wells, or the *well injectivity index* (WI) for injectors, and accounts for all rock and fluid properties, as well as geometric factors that affect the flow. In MRST, we do not distinguish between productivity and injectivity indices, and henceforth we will only use the shorthand WI.

Steady-State Flow in a Radial System

To derive the basic linear relation (4.27), we consider a vertical well that drains a rock with uniform permeability. The well penetrates the rock completely over a height h and is open in the radial direction. Gravity is neglected and the single-phase fluid is assumed to flow radially inward from an infinite domain; see Figure 4.5. From mass conservation written in cylinder coordinates, we have

$$\frac{1}{r}\frac{\partial(rvB^{-1})}{\partial r} = 0 \qquad \longrightarrow \qquad v = BC/r.$$



a) Horizontal view of well b) From above

Figure 4.5 Radial flow used to derive inflow–performance relationship.

https://doi.org/10.1017/9781108591416.007 Published online by Cambridge University Press

Here, the formation-volume factor B is the ratio between the volume of the fluid at reservoir conditions and the volume of the fluid at surface conditions.

The integration constant C is determined by integrating around a small cylinder surrounding the fluid sink,

$$q = \oint B^{-1}\vec{v} \cdot \vec{n}\, ds = -2\pi hC.$$

Inserting this into Darcy's law in cylinder coordinates, we have

$$v = -\frac{qB}{2\pi rh} = -\frac{K}{\mu}\frac{dp}{dr}.$$

Even if several different flow patterns can be expected when fluids flow toward a vertical wellbore, two-dimensional radial flow is considered to be the most representative for vertical oil and gas wells. We now integrate this equation from the wellbore r_w and to an arbitrary radius,

$$2\pi Kh \int_{p_w}^{p} \frac{1}{q\mu B}\, dp = \int_{r_w}^{r} \frac{1}{r}\, dr.$$

Here, B decreases with pressure and μ increases. The composite effect is that $(\mu B)^{-1}$ decreases almost linearly with pressure. We can therefore approximate μB by $(\mu B)_{avg}$ evaluated at the average pressure $(p_w + p)/2$, to obtain an explicit expression for the pressure (dropping the subscript on μB for brevity)

$$p(r) = p_w + \frac{q\mu B}{2\pi Kh} \ln(r/r_w). \tag{4.28}$$

Figure 4.6 shows a plot of this solution on a scale that is somewhat larger than the typical size of grid cells in field-scale models. Notice, in particular, that most of the pressure

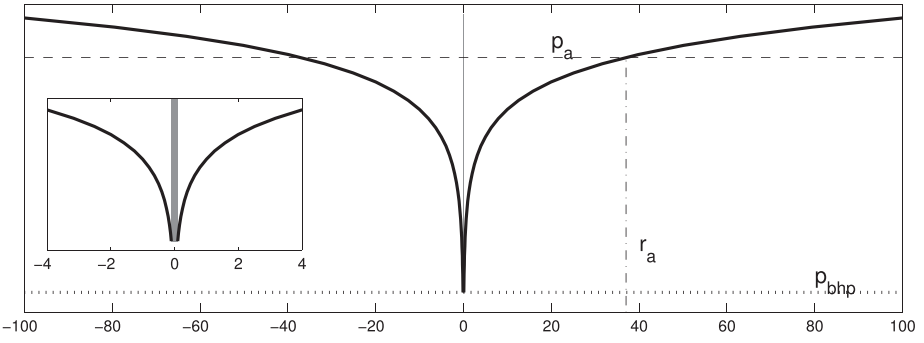


Figure 4.6 Illustration of the pressure distribution inside a circular region with a radius of 100 meters computed from (4.28), assuming the well is producing fluids from an infinite domain. Here, p_a is the volumetric pressure average inside the domain and r_a is the radius at which this value is found. The inset shows a zoom of the near-well zone.

https://doi.org/10.1017/9781108591416.007 Published online by Cambridge University Press

drawdown takes place within a relatively short distance to the wellbore, which confirms the need for analytical submodels to avoid having to introduce very fine grids near the wellbore.

Pseudo-Steady State in a Radial System

We can also derive a linear inflow relationship of the form (4.27) under somewhat more realistic conditions by considering *pseudo-steady* flow conditions. Assuming constant compressibility and fluid viscosity, the single-phase flow equation written in cylinder coordinates reads

$$\frac{1}{r}\frac{\partial}{\partial r}\left(r\frac{\partial p}{\partial r}\right) = \frac{\mu\phi c}{K}\frac{\partial p}{\partial t}. \tag{4.29}$$

At pseudo-steady state, $\partial p/\partial t$ is constant. To determine the pressure differential, we consider the global mass-balance, which states that the amount of produced fluid must equal the change of mass inside the reservoir. Writing this on integral form and differentiating gives us

$$\int_0^t q(t)\,dt = -\int_{p_0}^{\bar{p}} \frac{Ah\phi c}{B}\,d\bar{p} \qquad \longrightarrow \qquad \frac{\partial p}{\partial t} = \frac{d\bar{p}}{dt} = -\frac{B\,q}{\pi\left(r_d^2 - r_w^2\right)h\phi c}.$$

Here, q is the volumetric rate at surface conditions, p_0 is the initial reservoir pressure, and $\bar{p}$ is the average pressure inside the cylindrical drainage region at time t. In the expression above, we have introduced the formation-volume factor B, defined as the ratio between the volume of the fluid at reservoir conditions and the volume of the fluid at surface conditions. Inserting the constant pressure change into (4.29) gives

$$\frac{1}{r}\frac{d}{dr}\left(r\frac{dp}{dr}\right) = -\frac{q\mu B}{hK\pi\left(r_d^2 - r_w^2\right)} = -C.$$

To solve this equation, we first integrate with respect to r and use the fact that $dp/dr = 0$ at the outer boundary,

$$\int d\left(r\frac{dp}{dr}\right) = -\int Cr\,dr \qquad \overset{\frac{dp}{dr}|_{r_d}=0}{\longrightarrow} \qquad r\frac{dp}{dr} = \tfrac{1}{2}C\left(r_d^2 - r^2\right).$$

Then, we integrate outward from r_w to determine $p(r)$,

$$p(r) - p_w = \tfrac{1}{2}C\int_{r_w}^{r}\left(\frac{r_d^2}{r} - r\right)dr = \frac{q\mu B}{2\pi Kh}\left[\frac{r_d^2}{r_d^2 - r_w^2}\ln\left(\frac{r}{r_w}\right) - \frac{1}{2}\frac{r^2 - r_w^2}{r_d^2 - r_w^2}\right].$$

https://doi.org/10.1017/9781108591416.007 Published online by Cambridge University Press

This equation can be used to develop the desired relationship between the average reservoir pressure and the wellbore flowing pressure p_w,

$$\begin{aligned} p_a &= \frac{1}{V}\int_{r_w}^{r_d} p(r)\,dV = \frac{2\pi h\phi}{\pi(r_d^2 - r_w^2)h\phi}\int_{r_w}^{r_d} p(r) r\,dr \\ &= \frac{2}{r_d^2 - r_w^2}\int_{r_w}^{r_d}\left[p_w r + \frac{C}{2}\left(r_d^2 r\ln\left(\frac{r}{r_w}\right) - \tfrac{1}{2}r\left(r^2 - r_w^2\right)\right)\right] dr \\ &= p_w + \frac{2}{\left(r_d^2 - r_w^2\right)}\frac{q\mu B}{2\pi hK\left(r_d^2 - r_w^2\right)}\left[I_1 + I_2\right]. \end{aligned}$$

The two integrals can be computed as follows,

$$\begin{aligned} I_1 &= \int_{r_w}^{r_d} r\ln\left(r/r_w\right)\,dr = \tfrac{1}{2}r_d^2\ln\left(r_d/r_w\right) - \tfrac{1}{4}\left(r_d^2 - r_w\right)^2, \\ I_2 &= \int_{r_w}^{r_d}\tfrac{1}{2}r\left(r^2 - r_w^2\right)\,dr = \tfrac{1}{8}\left(r_d^4 - r_w^4\right) - \tfrac{1}{4}r_w^2\left(r_d^2 - r_w^2\right). \end{aligned}$$

We can now insert this back into the equation and use the fact that $r_d \gg r_w$ to derive our desired expression,

$$\begin{aligned} p_a - p_w &= \frac{q\mu B}{2\pi hK}\left[\underbrace{\frac{r_d^4}{(r_d^2 - r_w^2)^2}}_{\approx 1}\ln\left(\frac{r_d}{r_w}\right) - \frac{1}{2}\underbrace{\frac{r_d^2}{r_d^2 - r_w^2}}_{\approx 1} - \frac{1}{4}\underbrace{\frac{r_d^2 + r_w^2}{r_d^2 - r_w^2}}_{\approx 1} + \frac{1}{2}\underbrace{\frac{r_w^2}{r_d^2 - r_w^2}}_{\approx 0}\right] \\ &\approx \frac{q\mu B}{2\pi hK}\left[\ln\left(\frac{r_d}{r_w}\right) - \frac{3}{4}\right]. \end{aligned}$$

Rearranging terms, we obtain an expression for the fluid rate of the producer

$$q = \frac{2\pi Kh}{\mu B\left(\ln(r_d/r_w) - 0.75\right)}\left(p_a - p_w\right). \tag{4.30}$$

For an injector, we likewise obtain

$$q = \frac{2\pi Kh}{\mu B\left(\ln(r_d/r_w) - 0.75\right)}\left(p_w - p_a\right).$$

The relation (4.30) was developed for an ideal well under several simplifying assumptions: homogeneous and isotropic formation of constant thickness, clean wellbore, etc. A well will rarely experience these ideal conditions in practice. Typically, the permeability is altered close to the wellbore under drilling and completion, the well will only be partially completed, and so on. The actual pressure performance will therefore deviate from (4.30). To model this, it is customary to include a *skin factor* S to account for extra pressure loss due to alterations in the inflow zone. The resulting equation is

$$q = \frac{2\pi Kh}{\mu B\left(\ln(r_d/r_w) - 0.75 + S\right)}\left(p_a - p_w\right). \tag{4.31}$$

https://doi.org/10.1017/9781108591416.007 Published online by Cambridge University Press

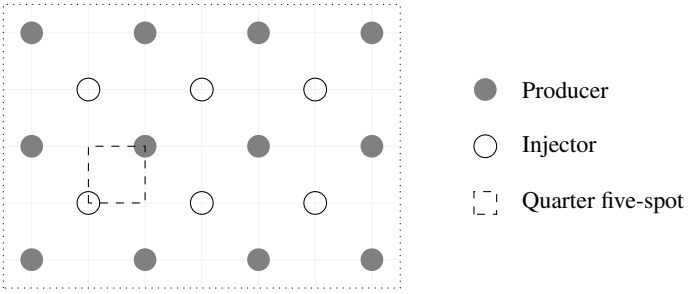


Figure 4.7 Excerpts of a repeated five-spot pattern.

Often the constant -0.75 is included in the skin factor S, which could be negative for stimulated wells. Sometimes h is modified to ch, where c is the completion factor, i.e., a dimensionless number between zero and one that describes the fraction of the wellbore open to flow.

Peaceman-Type Well Models

To use the radial model in conjunction with a reservoir model, the volumetric average pressure in the radial model must be related to the computed cell pressure p. Exact solutions are generally not known, since real reservoirs have complicated geometries and irregular boundaries. Well models are therefore developed using highly idealized reservoir geometries. One such example is the so-called *repeated five-spot pattern*, which consists of a thin, infinitely large, horizontal reservoir with a staggered pattern of injection and production wells as (see Figure 4.7) that repeats itself to infinity in all directions. The name comes from the fact that each injector is surrounded by four producers, and vice versa, hence creating tiles of five-spot patterns. If all wells operate at equal rates, the flow pattern has certain symmetries and it is common to only consider one quarter of the five spot, as shown in Figure 4.7, subject to no-flow boundary conditions. Muskat and Wyckoff [219] derived the following explicit solution for the pressure drop between the injection and production wells,

$$\Delta p = \frac{q\mu B}{\pi K h}\Big(\ln(d/r_w) - C\Big), \tag{4.32}$$

where d is the distance between the wells, and C is given by an infinite series. The authors originally used $C = 0.6190$, but a more accurate value, $C = 0.61738575$, was later derived by Peaceman [245], who used (4.32) to determine an equivalent radius r_e at which the cell pressure is equal to average of the exact pressure. Assuming isotropic permeabilities, square grid blocks, single-phase flow, and a well at a center of an interior block, Peaceman [247] showed that the equivalent radius is

$$r_e \approx 0.20788\sqrt{\Delta x\,\Delta y}$$

for the two-point discretization that will be discussed in more detail in Section 4.4.1. To give you an idea, let us present a somewhat simplified derivation in the incompressible case

https://doi.org/10.1017/9781108591416.007 Published online by Cambridge University Press

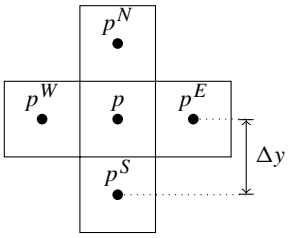


Figure 4.8 Five-point stencil on a grid with square grid blocks ($\Delta x = \Delta y$).

without gravity and assume that the viscosity and formation-volume factor are constant. For a single-layered square grid (Figure 4.8), the two-point scheme then simplifies to a standard five-point difference scheme

$$-\frac{Kh}{\mu B}\left[4p - p^W - p^N - p^E - p^S\right] = q.$$

The solution is symmetric for the repeated five-spot pattern, so that $p^E = p^N = p^W = p^S$ (which explains why μB could be factored out). Hence, we can eliminate the last three pressure values and write the five-point stencil as

$$p = p^E - \frac{q\mu B}{4Kh}.$$

Next, we assume that the radial well solution is an accurate approximation to the pressure values in the neighboring cells,

$$p^E = p_w + \frac{q\mu B}{2\pi Kh}\ln\left(\Delta x/r_w\right).$$

Combining the two equations, we have

$$\begin{aligned} p &= p_w + \frac{q\mu B}{2\pi Kh}\ln\left(\Delta x/r_w\right) - \frac{q\mu B}{4Kh} \\ &= p_w + \frac{q\mu B}{2\pi Kh}\left[\ln\left(\Delta x/r_w\right) - \frac{\pi}{2}\right] = p_w + \frac{q\mu B}{2\pi Kh}\ln\left(e^{-\pi/2}\Delta x/r_w\right). \end{aligned}$$

Now, $\exp(-\pi/2) \approx 0.20788$, and hence we have derived Peaceman's classical result.

This basic model has later been extended to cover a lot of other cases, e.g., off-center wells, multiple wells, non-square grids, anisotropic permeability, horizontal wells; see for instance [18, 102, 13]. For anisotropic permeabilities $\mathbf{K}$ =diag(K_x, K_y, K_z), the permeability is replaced by an effective permeability

$$K_e = \sqrt{K_x\, K_y}, \tag{4.33}$$

https://doi.org/10.1017/9781108591416.007 Published online by Cambridge University Press

which follows by a coordinate transformation $x \to \hat{x}\sqrt{K_x}$, etc. For a grid with rectangular grid blocks ($\Delta x \neq \Delta y$), the equivalent radius reads [245]

$$r_e = 0.28 \frac{\left(\sqrt{K_y/K_x}\Delta x^2 + \sqrt{K_x/K_y}\Delta y^2\right)^{1/2}}{\left(K_y/K_x\right)^{1/4} + \left(K_x/K_y\right)^{1/4}}. \tag{4.34}$$

The same formulas apply to horizontal wells, with the necessary permutations; that is, for a well in the x-direction $\Delta x \to \Delta z$ and $K_x \to K_z$.

If we include gravity forces in the well and assume hydrostatic equilibrium, the well model reads

$$q_i = \frac{2\pi h K_e}{\mu_i B_i [\ln(r_e/r_w) + S]} \Big(p_R - p_w - \rho_i (z - z_{bh}) g \Big), \tag{4.35}$$

where K_e is given by (4.33) and r_e is given by (4.34). For deviated wells, h denotes the length of the grid block in the major direction of the wellbore and *not* the length of the wellbore.

At this point we should add a word of caution. The equivalent radius of a numerical method generally depends on how the method approximates pressure inside the grid cell containing the well perforation. The formulas given in this subsection are, strictly speaking, only valid if you use the specific two-point discretization they were developed for. When using another discretization method, you may have to compute other values for the equivalent radius, e.g., as discussed in [192, 197].

4.3.3 Field Lines and Time-of-Flight

Equation (4.10) together with a set of suitable and compatible boundary conditions constitute all you need to determine the pressure distribution and flow velocity of an incompressible fluid inside an incompressible rock. In the remainder of this section, we will discuss a few simple concepts and auxiliary equations you can use to visualize, analyze, and improve your understanding of the computed flow fields.

A simple way to visualize a flow field is to use field lines resulting from the vector field: streamlines, streaklines, and pathlines. In steady flow, the three are identical. However, if the flow is not steady, i.e., when $\vec{v}$ changes with time, they differ. *Streamlines* are associated with an instant snapshot of the flow field and consist of a family of curves that are everywhere tangential to $\vec{v}$ and show the direction a fluid element will travel at this specific point in time. That is, if $\vec{x}(r)$ is a parametric representation of a single streamline at this instance $\hat{t}$ in time, then

$$\frac{d\vec{x}}{dr} \times \vec{v}(\vec{x}, \hat{t}) = 0, \quad \text{or equivalently,} \quad \frac{d\vec{x}}{dr} = \frac{\vec{v}(\hat{t})}{|\vec{v}(\hat{t})|}. \tag{4.36}$$

In other words, streamlines are calculated throughout the fluid from an *instantaneous* snapshot of the flow field. Because two streamlines from the same instance in time cannot cross, there cannot be flow across a streamline, and if we align a coordinate along a bundle of streamlines, the flow through them will be one-dimensional.

https://doi.org/10.1017/9781108591416.007 Published online by Cambridge University Press

Pathlines are the trajectories individual fluid elements will follow over a certain period. In each moment of time, the path a fluid particle takes will be determined by the streamlines associated with the velocity field at this instance in time. If $\vec{y}(t)$ represents a single path line starting in $\vec{y}_0$ at time t_0, then

$$\frac{d\vec{y}}{dt} = \vec{v}(\vec{y}, t), \qquad \vec{y}(t_0) = \vec{y}_0. \tag{4.37}$$

A *streakline* is the line traced out by all fluid particles that have passed through a prescribed point throughout a certain period of time. (Think of dye injected into the fluid at a specific point.) If we let $\vec{z}(t, s)$ denote a parametrization of a streakline and $\vec{z}_0$ the specific point through which all fluid particles have passed, then

$$\frac{d\vec{z}}{dt} = \vec{v}(\vec{z}, t), \qquad \vec{z}(s) = \vec{z}_0. \tag{4.38}$$

Like streamlines, two streaklines cannot intersect each other. In summary: *streamline patterns change over time, but are easy to generate mathematically. Pathlines and streaklines are recordings of the passage of time obtained through experiments.*

Streamlines are far more used within reservoir simulation than pathlines and streaklines. Instead of using the arc length r to parametrize streamlines, it is common to introduce an alternative parametrization called *time-of-flight*, which takes into account the reduced volume available for flow, i.e., the porosity ϕ. Time-of-flight is defined by the following integral

$$\tau(r) = \int_0^r \frac{\phi(\vec{x}(s))}{|\vec{v}(\vec{x}(s))|}\, ds, \tag{4.39}$$

where τ expresses the time it takes a fluid particle to travel a distance r along a streamline (in the interstitial velocity field $\vec{v}/\phi$). Alternatively, computing the directional derivative of τ along a streamline and applying the fundamental theorem of calculus to (4.39),

$$\left(\frac{\vec{v}}{|\vec{v}|} \cdot \nabla\right) \tau = \frac{d}{dr}\, \tau = \frac{\phi}{|\vec{v}|}$$

from which it follows that τ can be expressed by a differential equation [78, 79]

$$\vec{v} \cdot \nabla \tau = \phi. \tag{4.40}$$

For lack of a better name, we will refer to this as the *time-of-flight equation*.

4.3.4 Tracers and Volume Partitions

A tracer can somewhat simplified be seen as a neutral particle that passively flow with the fluid without altering its flow properties. The concentration of a tracer is given by a continuity equation on the same form as (4.5),

$$\frac{\partial(\phi C)}{\partial t} + \nabla \cdot (\vec{v} C) = q_C. \tag{4.41}$$

https://doi.org/10.1017/9781108591416.007 Published online by Cambridge University Press

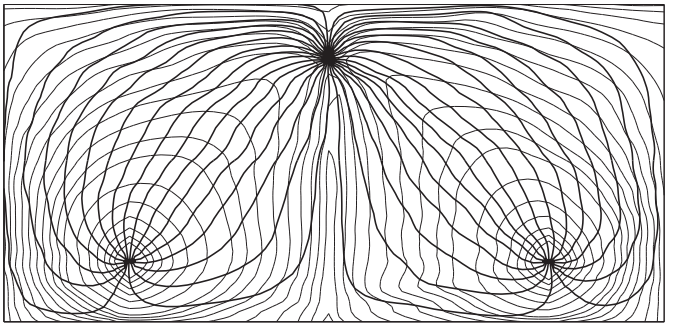


Figure 4.9 Illustration of time-of-flight, shown as gray isocontour lines, and streamlines shown as thick black lines.

Volumetric connections within a reservoir can be determined by simulating the evolution of artificial, nondispersive and nondiffusive tracers whose concentration does not change upon fluid compression or expansion. A simple flow diagnostic is to set the tracer concentration equal to one in a particular fluid source or at a certain part of the inflow boundary, and compute the solution approached at steady-state conditions from the nonconservative equation,

$$\vec{v} \cdot \nabla C = q_C, \qquad C|_{\text{inflow}} = 1. \tag{4.42}$$

The resulting *tracer distribution* gives the portion of the total fluid volume passing through a point in the reservoir that can be attributed to a given fluid source or parts of the inflow boundary. Likewise, by reversing the sign of the flow field and assigning unit tracers to a particular fluid sink or parts of the outflow, one can compute the portion of the fluid content at a certain point in the reservoir that eventually will arrive at the given fluid sink or outflow boundary. By repeating this process for all parts of the inflow, we obtain a partition of the instantaneous flow field: Let C_i denote the tracer associated with fluid source Q_i. This fluid source will influence the flow through all points $\vec{x}$ for which $C_i(\vec{x}) > 0$. If $C(\vec{x}) = 1$, then Q_i is the only fluid source affecting the flow through $\vec{x}$. Otherwise, the point must be affected by at least one other fluid source Q_j. The tracer distributions $\{C_i(\vec{x})\}_{i=1}^m$ can thus be seen as measures of how strongly the flow through a point is influenced by various fluid sources. We will return with detailed discussion in Chapter 13 on flow diagnostics.

You can also use streamlines and time-of-flight to define an alternative curvilinear and flow-based coordinate system in three dimensions. To this end, we introduce the bi-streamfunctions ψ and χ [34], for which $\vec{v} = \nabla\psi \times \nabla\chi$. In the streamline coordinates (τ, ψ, χ), the gradient operator is expressed as

$$\nabla_{(\tau,\psi,\chi)} = (\nabla\tau)\frac{\partial}{\partial\tau} + (\nabla\psi)\frac{\partial}{\partial\psi} + (\nabla\chi)\frac{\partial}{\partial\chi}. \tag{4.43}$$

Moreover, a streamline Ψ is defined by the intersection of a constant value for ψ and a constant value for χ. Because $\vec{v}$ is orthogonal to $\nabla\psi$ and $\nabla\chi$, it follows from (4.40) that

https://doi.org/10.1017/9781108591416.007 Published online by Cambridge University Press

$$\vec{v} \cdot \nabla_{(\tau,\psi,\chi)} = (\vec{v} \cdot \nabla\tau)\frac{\partial}{\partial\tau} = \phi\frac{\partial}{\partial\tau}. \tag{4.44}$$

Therefore, the coordinate transformation $(x, y, z) \rightarrow (\tau, \psi, \chi)$ will reduce the three-dimensional transport equation (4.41) to a family of one-dimensional transport equations along each streamline [78, 164]. For incompressible flow this reads

$$\frac{\partial C}{\partial t} + \frac{\partial C}{\partial \tau} = 0. \tag{4.45}$$

There is no exchange of the quantity C among streamlines, and we can thus view each streamline as an isolated flow system. Assuming a prescribed concentration history $C_0(t)$ at the inflow gives a time-dependent boundary-value problem for the concentration at the outflow. Here, the response is given as (see [78]),

$$C(t) = C_0(t - \tau), \tag{4.46}$$

which is easily verified by inserting the expression into (4.45) and using the fact that the solution is unique [134]. For the special case of continuous and constant injection, the solution is particularly simple

$$C(t) = \begin{cases} 0, & t < \tau, \\ C_0, & t > \tau. \end{cases}$$

4.4 Basic Finite-Volume Discretizations

Extensive research on numerical methods for the Laplace/Poisson equation has led to a large number of different finite-difference and finite-volume methods, as well as finite-element methods based on standard, mixed, or discontinuous Galerkin formulations. In the rest of this section we present the simplest example of a finite-volume discretization, the two-point flux-approximation scheme, which is used extensively throughout industry and also is the default discretization method in MRST. We give a detailed derivation of the two-point method and briefly outline how to discretize the time-of-flight and the stationary tracer equations.

The two-point method is simple to implement, gives sparse linear systems that are not too costly to invert, and is robust in the sense that it produces monotone pressure approximations. Unfortunately, the method is only conditionally consistent, and will introduce numerical artifacts and not provide convergent solutions for increased grid resolution if the grid does not satisfy a certain orthogonality condition with respect to the permeability tensor. We discuss this in more detail in Chapter 6, which also introduces more advanced and consistent discretizations that are guaranteed to be convergent on general polyhedral grids and for anisotropic permeabilities.

4.4.1 Two-Point Flux-Approximation

To keep technical details at a minimum, we will in the following without loss of generality consider the simplified single-phase flow equation

https://doi.org/10.1017/9781108591416.007 Published online by Cambridge University Press

$$\nabla \cdot \vec{v} = q, \qquad \vec{v} = -\mathbf{K}\nabla p, \qquad \text{in } \Omega \subset \mathbb{R}^d. \tag{4.47}$$

In classical finite-difference methods, partial differential equations are approximated by replacing the derivatives with appropriate divided differences between point-values on a discrete set of points in the domain. Finite-volume methods, on the other hand, have a more physical motivation and are derived from conservation of (physical) quantities over cell volumes. Thus, in a finite-volume method the unknown quantities are represented in terms of average values defined over a set of finite volumes. The PDE model is integrated and required to hold in an averaged sense rather than in a pointwise sense. In this sense, finite-difference and finite-volume methods have fundamentally different interpretation and derivation, but the names are often used interchangeably in the scientific literature. The main reason for this is probably that the discrete equations derived for the cell-centered values in a low-order, mass-conservative, finite-difference method are identical to the discrete equations for the cell averages in corresponding finite-volume methods. Herein, we will stick to this convention and not make a strict distinction between the two types of methods.

To develop a finite-volume discretization for (4.47), we start by rewriting the equation in integral form using a single cell Ω_i in the discrete grid as control volume

$$\int_{\partial\Omega_i} \vec{v} \cdot \vec{n}\, ds = \int_{\Omega_i} q\, d\vec{x}. \tag{4.48}$$

This is a simpler form of (4.3), where the accumulation term has disappeared because ϕ and ρ are independent of time and the constant ρ has been eliminated. Equation (4.48) ensures that mass is conserved for each grid cell. The next step is to use Darcy's law to compute the flux across each face of the cell,

$$v_{i,k} = \int_{\Gamma_{ik}} \vec{v} \cdot \vec{n}\, ds, \qquad \Gamma_{i,k} = \partial\Omega_i \cap \partial\Omega_k. \tag{4.49}$$

We refer to the faces $\Gamma_{i,k}$ as *half-faces*, since they are associated with a particular grid cell Ω_i and a certain normal vector $\vec{n}_{i,k}$. However, since the grid is assumed to be matching, each interior half face will have a twin half-face $\Gamma_{k,i}$ that has identical area $A_{k,i} = A_{i,k}$ but opposite normal vector $\vec{n}_{k,i} = -\vec{n}_{i,k}$. If we now approximate the integral over the cell face in (4.49) by the midpoint rule, we can use Darcy's law to write the flux as

$$v_{i,k} \approx A_{i,k}\vec{v}(\vec{x}_{i,k}) \cdot \vec{n}_{i,k} = -A_{i,k}\,(\mathbf{K}\nabla p)\,(\vec{x}_{i,k}) \cdot \vec{n}_{i,k}, \tag{4.50}$$

where $\vec{x}_{i,k}$ denotes the centroid on $\Gamma_{i,k}$. The idea is now to use a one-sided finite difference to express the pressure gradient as the difference between the pressure $\pi_{i,k}$ at the face centroid and the pressure at some point inside the cell. However, in a finite-volume method, we only know the cell averaged value of the pressure inside the cell. We must therefore make some additional assumption to be able to reconstruct the point values we need to estimate the pressure gradient in Darcy's law. If we assume that the pressure is linear (or constant) inside each cell, the reconstructed pressure value at the cell center is identical to the average pressure p_i inside the cell, and hence it follows that (see Figure 4.10)

$$v_{i,k} \approx A_{i,k}\mathbf{K}_i \frac{(p_i - \pi_{i,k})\vec{c}_{i,k}}{|\vec{c}_{i,k}|^2} \cdot \vec{n}_{i,k} = T_{i,k}(p_i - \pi_{i,k}). \tag{4.51}$$

https://doi.org/10.1017/9781108591416.007 Published online by Cambridge University Press

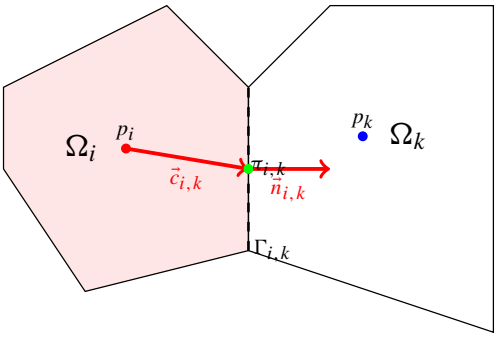


Figure 4.10 Two cells used to define the two-point finite-volume discretization of the Laplace operator.

Here, we have introduced one-sided *transmissibilities* $T_{i,k}$ that are associated with a single cell and give a two-point relation between the flux across a cell face and the difference between the pressure at the cell and face centroids. We refer to these one-sided transmissibilities as *half-transmissibilities*, since they are associated with a half face.

To derive the final discretization, we impose continuity of fluxes across all faces, $v_{i,k} = -v_{k,i} = v_{ik}$ and continuity of face pressures $\pi_{i,k} = \pi_{k,i} = \pi_{ik}$. This gives us two equations,

$$T_{i,k}^{-1} v_{ik} = p_i - \pi_{ik}, \qquad -T_{k,i}^{-1} v_{ik} = p_k - \pi_{ik}.$$

By eliminating the interface pressure π_{ik}, we end up with the following two-point flux-approximation (TPFA) scheme,

$$v_{ik} = \left[T_{i,k}^{-1} + T_{k,i}^{-1} \right]^{-1} (p_i - p_k) = T_{ik}(p_i - p_k), \tag{4.52}$$

where T_{ik} is the transmissibility associated with the connection between the two cells. As the name suggests, the TPFA scheme uses two "points," the cell averages p_i and p_k, to approximate the flux across the interface Γ_{ik} between cells Ω_i and Ω_k. In our derivation so far, we have parametrized the cell fluxes in terms of the index of the neighboring cell. Extending the derivation to also include fluxes on exterior faces is trivial since we either know the flux explicitly for Neumann boundary conditions (4.23) or (4.25), or know the interface pressure for Dirichlet conditions (4.24).

By inserting the expression for v_{ik} into (4.48), we see that the TPFA scheme for (4.47), in compact form, seeks a set of cell averages that satisfy the following system of equations

$$\sum_k T_{ik}(p_i - p_k) = q_i, \qquad \forall \Omega_i \subset \Omega. \tag{4.53}$$

This system is clearly symmetric, and a solution is only defined up to an arbitrary constant, as is also the case for the corresponding Poisson problem. We can make the system positive-definite and preserve symmetry by specifying the pressure in a single point. In MRST, we

https://doi.org/10.1017/9781108591416.007 Published online by Cambridge University Press

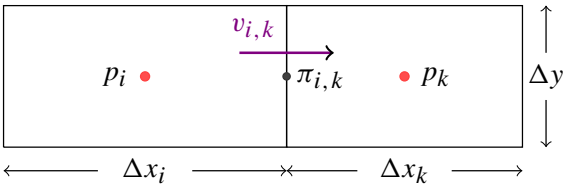


Figure 4.11 Two cells used to derive the TPFA discretization for a 2D Cartesian grid.

have chosen to set $p_1 = 0$ by adding a positive constant to the first diagonal of the matrix $\mathbf{A} = [a_{ij}]$, where:

$$a_{ij} = \begin{cases} \sum_k T_{ik} & \text{if } j = i, \\ -T_{ij} & \text{if } j \neq i. \end{cases}$$

The matrix $\mathbf{A}$ is sparse and will have a banded structure for structured grids: tridiagonal for 1D grids and penta- and heptadiagonal for logically Cartesian grids in 2D and 3D, respectively. The TPFA scheme is monotone, robust, and relatively simple to implement, and is currently the industry standard with reservoir simulation.

Example 4.4.1 *To tie the links with standard finite-difference methods on Cartesian grids, we derive the two-point discretization for a 2D Cartesian grid with isotropic permeability. Consider the flux in the* x*-direction between two cells* i *and* k *as illustrated in Figure 4.11. As above, we impose mass conservation inside each cell. For cell* i *this reads:*

$$v_{i,k} = \Delta y \frac{(p_i - \pi_{i,k})}{\left(\frac{1}{2}\Delta x_i\right)^2} \left(\tfrac{1}{2}\Delta x_i, 0\right) K_i \, (1,0)^T = \Delta y \frac{2K_i}{\Delta x_i} \left(p_i - \pi_{i,k}\right)$$

and likewise for cell k*:*

$$v_{k,i} = \Delta y \frac{(p_k - \pi_{k,i})}{\left(\frac{1}{2}\Delta x_k\right)^2} \left(-\tfrac{1}{2}\Delta x_k, 0\right) K_k \, (-1,0)^T = \Delta y \frac{2K_k}{\Delta x_k} \left(p_k - \pi_{k,i}\right)$$

Next, we impose continuity of fluxes and face pressures,

$$v_{i,k} = -v_{k,i} = v_{ik}, \qquad \pi_{i,k} = \pi_{k,i} = \pi_{ik},$$

which gives us two equations

$$\frac{\Delta x_i}{2K_i \Delta y} v_{ik} = p_i - \pi_{ik}, \qquad -\frac{\Delta x_k}{2K_k \Delta y} v_{ik} = p_k - \pi_{ik}.$$

Finally, we eliminate π_{ik} *to obtain*

$$v_{ik} = 2\Delta y \left(\frac{\Delta x_i}{K_i} + \frac{\Delta x_k}{K_k}\right)^{-1} (p_i - p_k),$$

which shows that the transmissibility is given by the harmonic average of the permeability values in the two adjacent cells, as one would expect.

https://doi.org/10.1017/9781108591416.007 Published online by Cambridge University Press

In [2], two former colleagues and I showed how you can implement an efficient and self-contained MATLAB program that in approximately 30 compact lines solves the incompressible flow equation (4.47) using the two-point method outlined in this section. The program was designed for Cartesian grids with no-flow boundary conditions only, and relied strongly on the logical ijk numbering of grid cells. For this reason, the program has limited applicability beyond highly idealized cases such as the SPE 10 model. However, in its simplicity, it presents an interesting contrast to the general-purpose implementation in MRST which is designed to handle unstructured grids, wells, and more general boundary conditions. I encourage you to read the paper and try the accompanying program and example scripts, which you can download from my personal website.

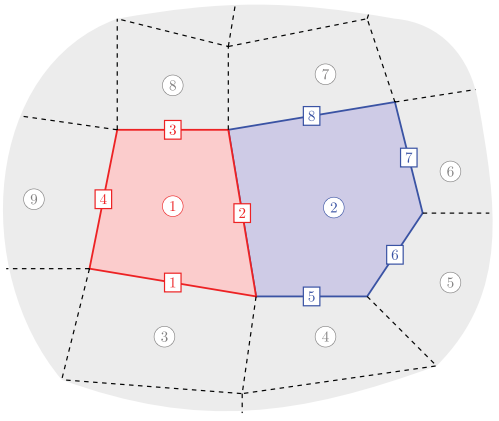
4.4.2 Discrete `div` and `grad` Operators

The double-index notation $v_{i,k}$ and v_{ik} used in the previous section is simple and easy to comprehend when working with a single interface between two neighboring cells. It becomes more involved when we want to introduce the same type of discretizations for more complex models than the Poisson equation for incompressible flow. To prepare for discussions later in the book, we will now introduce a more abstract way of writing the two-point finite-volume discretization. The idea is to introduce discrete operators for the divergence and gradient operators that mimic their continuous counterparts. This will enable us to write the discretized version of Poisson's equation (4.47) in the same form as its continuous counterpart. These operators will necessarily have to incorporate information about the grid topology in the same way as the transmissibility incorporates information about grid geometry. However, once defined, we can apply them to any quantities defined over the cells and faces of a grid without knowing anything about the underlying grid topology. This gives us a powerful tool to write solvers in a way that effectively is grid agnostic.

We start by a quick recap of the definition of unstructured grids. As discussed in detail in Section 3.4, the grid structure in MRST consists of three objects: *cells*, *faces*, and *nodes*. Each cell is delimited by a set of faces, and each face by a set of *edges*, which again are determined by a pair of nodes. Each object has given geometrical properties (volume, areas, centroids). As before, let n_c and n_f denote the number of cells and faces respectively. To define the topology of the grid, we can use two mappings. The first is given by $F : \{1, \ldots, n_c\} \rightarrow \{0, 1\}^{n_f}$ and maps a cell to the set of faces that delimit this cell. In the grid structure `G`, this is represented as the `G.cells.faces` array, where the first column specifying the cell numbers is not stored, since it is redundant and instead must be computed by a call `f2cn = gridCellNo(G)`. The second mapping consists of two components that for a given face give the corresponding neighboring cells, $C_1, C_2 : \{1, \ldots, n_f\} \rightarrow \{1, \ldots, n_c\}$. In the grid structure `G`, C_1 is given by `G.faces.neighbors(:,1)` and C_2 by `G.faces.neighbors(:,2)`. This is illustrated in Figure 4.12.

Let us now construct the discrete analogues of the divergence and gradient operators, which we denote `div` and `grad`. The divergence operator `div` is a linear mapping from

https://doi.org/10.1017/9781108591416.007 Published online by Cambridge University Press



c	F(c)
1	1
1	2
1	3
1	4
2	5
2	6
2	7
2	8
2	2
3	1
⋮	⋮
⋮	⋮

f	C_1	C_2
1	3	1
2	1	2
3	1	8
4	9	1
5	4	2
6	2	5
7	2	6
8	2	7
⋮	⋮	⋮
⋮	⋮	⋮

Figure 4.12 Illustration of the mappings from cells to faces and from faces to cells used to define the discrete divergence and gradient operators.

faces to cells, which typically will be applied to a discrete flux $\boldsymbol{v} \in \mathbb{R}^{n_f}$. We write $\boldsymbol{v}[f]$ to denote the flux evaluated on a face f and assume that the orientation of this flux is from $C_1(f)$ to $C_2(f)$. Hence, the total amount of matter leaving a cell c is given by the sum of outfluxes minus the sum of influxes

$$\mathtt{div}(\boldsymbol{v})[c] = \sum_{f \in F(c)} \boldsymbol{v}[f]\, \mathbf{1}_{\{c=C_1(f)\}} - \sum_{f \in F(c)} \boldsymbol{v}[f]\, \mathbf{1}_{\{c=C_2(f)\}}. \tag{4.54}$$

Here, element c of the vector $\mathbf{1}_{\{c=C_1(f)\}}$ equals one if cell c is neighbor to face f and zero otherwise. The `grad` operator maps $\mathbb{R}^{n_c}$ to $\mathbb{R}^{n_f}$ and is defined as

$$\mathtt{grad}(\boldsymbol{p})[f] = \boldsymbol{p}[C_2(f)] - \boldsymbol{p}[C_1(f)], \tag{4.55}$$

for any $\boldsymbol{p} \in \mathbb{R}^{n_c}$. In the continuous case, the gradient operator is the adjoint of the divergence operator (up to a sign), as we have

$$\int_\Omega p\, \nabla \cdot \vec{v}\, d\vec{x} + \int_\Omega \vec{v} \cdot \nabla p\, d\vec{x} = 0, \tag{4.56}$$

for vanishing boundary conditions. Let us prove that this property also holds in the discrete case. To simplify notation, we set $S_c = \{1, \ldots, n_c\}$ and $S_f = \{1, \ldots, n_f\}$. For any $\boldsymbol{v} \in \mathbb{R}^{n_f}$ and $\boldsymbol{p} \in \mathbb{R}^{n_c}$, we have

$$\begin{aligned}
\sum_{c \in S_c} \mathtt{div}(\boldsymbol{v})[c]\, \boldsymbol{p}[c] &= \sum_{c \in S_c} \boldsymbol{p}[c] \Big(\sum_{f \in F(c)} \boldsymbol{v}[f]\, \mathbf{1}_{\{c=C_1(f)\}} - \sum_{f \in F(c)} \boldsymbol{v}[f]\, \mathbf{1}_{\{c=C_2(f)\}} \Big) \\
&= \sum_{c \in S_c} \sum_{f \in S_f} \boldsymbol{v}[f]\, \boldsymbol{p}[c]\, \mathbf{1}_{\{c=C_1(f)\}}\, \mathbf{1}_{\{f \in F(c)\}} \\
&\quad - \sum_{c \in S_c} \sum_{f \in S_f} \boldsymbol{v}[f]\, \boldsymbol{p}[c]\, \mathbf{1}_{\{c=C_2(f)\}}\, \mathbf{1}_{\{f \in F(c)\}}.
\end{aligned} \tag{4.57}$$

https://doi.org/10.1017/9781108591416.007 Published online by Cambridge University Press

We can switch the order in the sums above and obtain

$$\sum_{c\in S_c}\sum_{f\in S_f} \boldsymbol{v}[f]\,\boldsymbol{p}[c]\mathbf{1}_{\{c=C_1(f)\}}\,\mathbf{1}_{\{f\in F(c)\}} = \sum_{f\in S_f}\sum_{c\in S_c} \boldsymbol{v}[f]\,\boldsymbol{p}[c]\,\mathbf{1}_{\{c=C_1(f)\}}\,\mathbf{1}_{\{f\in F(c)\}}.$$

For a given face f, we have that $\mathbf{1}_{\{c=C_1(f)\}}\mathbf{1}_{\{f\in F(c)\}}$ is nonzero if and only if $c = C_1(f)$, and therefore

$$\sum_{f\in S_f}\sum_{c\in S_c} \mathbf{1}_{\{c=C_1(f)\}}\mathbf{1}_{\{f\in F(c)\}}\boldsymbol{v}[f]\boldsymbol{p}[c] = \sum_{f\in S_f} \boldsymbol{v}[f]\boldsymbol{p}[C_1(f)].$$

In the same way, we have

$$\sum_{c\in S_c}\sum_{f\in S_f} \boldsymbol{v}[f]\,\boldsymbol{p}[c]\,\mathbf{1}_{\{c=C_2(f)\}}\,\mathbf{1}_{\{f\in F(c)\}} = \sum_{f\in S_f} \boldsymbol{v}[f]\,\boldsymbol{p}[C_2(f)],$$

so that (4.57) yields

$$\sum_{c\in S_c} \mathtt{div}(\boldsymbol{v})[c]\,\boldsymbol{p}[c] + \sum_{f\in S_f} \mathtt{grad}(\boldsymbol{p})[f]\,\boldsymbol{v}[f] = 0. \tag{4.58}$$

Until now, we have ignored boundary conditions. We can include them by introducing one more cell number $c = 0$ to denote the exterior. Then, we can consider external faces and extend the mappings C_1 and C_2 to $S_c \cup \{0\}$ so that a given face f is external if it satisfies $C_1(f) = 0$ or $C_2(f) = 0$. Note that the `grad` operator only defines values on internal faces. Now taking external faces into account, we obtain

$$\begin{aligned}\sum_{c\in S_c} \mathtt{div}(\boldsymbol{v})[c]\,\boldsymbol{p}[c] + \sum_{f\in S_f} \mathtt{grad}(\boldsymbol{p})[f]\,\boldsymbol{v}[f] \\ = \sum_{f\in \overline{S}_f\setminus S_f} \Big(\boldsymbol{p}[C_1(f)]\,\mathbf{1}_{\{C_2(f)=0\}} - \boldsymbol{p}[C_2(f)]\,\mathbf{1}_{\{C_1(f)=0\}}\Big)\,\boldsymbol{v}[f],\end{aligned} \tag{4.59}$$

where $\overline{S}_f$ denotes the extended set of faces, consisting of both internal and external faces. Identity (4.59) is the discrete counterpart to

$$\int_\Omega p\nabla\cdot\vec{v}\,d\vec{x} + \int_\Omega \vec{v}\cdot\nabla p\,d\vec{x} = \int_{\partial\Omega} p\vec{v}\cdot\vec{n}\,ds. \tag{4.60}$$

Let us now use these operators to discretize the Poisson equation. Going back to (4.47), let the vector $\boldsymbol{v} \in \mathbb{R}^{n_f}$ be a discrete approximation of the flux on faces; that is, for $f \in S_f$, we have

$$\boldsymbol{v}[f] \approx \int_{\Gamma_f} \vec{v}(x)\cdot\vec{n}_f\,ds,$$

https://doi.org/10.1017/9781108591416.007 Published online by Cambridge University Press

where $\vec{n}_f$ is the normal to face f with orientation given by the grid. The relation between the discrete pressure $\boldsymbol{p} \in \mathbb{R}^{n_c}$ and the discrete flux is given by the two-point flux approximation discussed in the previous section,

$$\boldsymbol{v}[f] = -\boldsymbol{T}[f]\,\mathtt{grad}(\boldsymbol{p})[f] \approx -\int_{\Gamma_f} \mathbf{K}(x)\nabla p \cdot \vec{n}_f \, ds. \tag{4.61}$$

Here, $\boldsymbol{T}[f]$ denotes the transmissibility associated with face f, as defined in (4.52). Hence, the discretization of (4.47) can be written as

$$\mathtt{div}(\boldsymbol{v}) = q \tag{4.62a}$$

$$v = -T\,\mathtt{grad}(\boldsymbol{p}), \tag{4.62b}$$

where the multiplication in (4.62b) holds element-wise.

We end our discussion by two examples that demonstrate how these discrete operators can be used to write very compact solvers that easily can be extended to solve problems of increasing complexity with minimal modifications to the code.

Example 4.4.2 *To illustrate the use of the discrete operators, let us set up and solve the classical Poisson equation on a simple box geometry,*

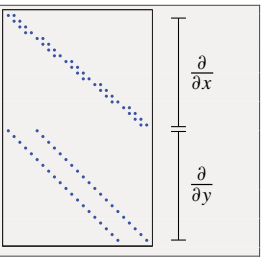
$$-\mathtt{div}(T\,\mathtt{grad}(\boldsymbol{p})) = \boldsymbol{q}, \qquad \Omega = [0,1] \times [0,1], \tag{4.63}$$

subject to no-flow boundary conditions with q consisting of a point source at (0,0) and a point sink at (1,1). First, we construct a small Cartesian grid

```
G = computeGeometry(cartGrid([5 5],[1 1]));
```

for which T equals a scalar multiple of the identity matrix and is therefore dropped for simplicity. The natural way to represent the `div` *and* `grad` *operators is to use sparse matrices. To this end, we use (4.58), which implies that the sparse matrix representing* `div` *is the negative transpose of the matrix that defines* `grad` *using (4.55). Moreover, since we assume no-flow boundary conditions, we only need to let C_1 and C_2 account for internal connections:*

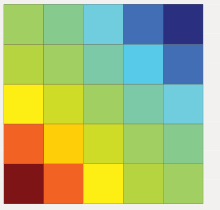
```
C = G.faces.neighbors;
C = C(all(C ~= 0, 2), :);
nf = size(C,1);
nc = G.cells.num;
D = sparse([(1:nf)'; (1:nf)'], C, ...
           ones(nf,1)*[-1 1], nf, nc);
grad = @(x) D*x;
div  = @(x) -D'*x;
```

$D =$ 

Once we have the discrete operators, we can write (4.63) in residual form, $\boldsymbol{f}(\boldsymbol{p}) = \boldsymbol{A}\boldsymbol{p} + q = 0$ and use automatic differentiation as discussed in Example A.5.3 to obtain $\boldsymbol{A}$ by computing $\partial \boldsymbol{f}/\partial \boldsymbol{p}$

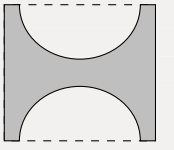
https://doi.org/10.1017/9781108591416.007 Published online by Cambridge University Press

```
p = initVariablesADI(zeros(nc,1));
q = zeros(nc, 1);            % source term
q(1) = 1; q(nc) = -1;        % -> quarter five-spot

eq    = div(grad(p))+q;      % equation
eq(1) = eq(1) + p(1);        % make solution unique
p     = -eq.jac{1}\eq.val;   % solve equation
```



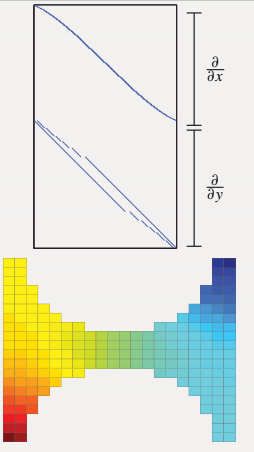
Next, we try to solve the same type of flow problem on a non-rectangular domain. That is, we still consider the unit square, but remove two half-circles of radius 0.4 centered at (0.5,0) and (0.5,1) respectively. To construct the corresponding grid, we use the fictitious grid approach from Section 3.1 (see Exercise 3.1.1):

```
G = cartGrid([20 20],[1 1]);
G = computeGeometry(G);
r1 = sum(bsxfun(@minus,G.cells.centroids,[0.5 1]).^2,2);
r2 = sum(bsxfun(@minus,G.cells.centroids,[0.5 0]).^2,2);
G = extractSubgrid(G, (r1>0.16) & (r2>0.16));
```



The construction of the discrete operators is agnostic to the exact layout of the grid, and because the transmissibility matrix T is still a multiple of the identity matrix, since the grid cells are equidistant squares, we can simply reuse the exact same set-up as in the previous code box:

```
% Grid information
C = G.faces.neighbors;
:
% Operators
D = sparse([(1:nf)'; (1:nf)'], C, ...
           ones(nf,1)*[-1 1], nf, nc);
:
% Assemble and solve equations
p = initVariablesADI(zeros(nc,1));
q = zeros(nc, 1);
q(1) = 1; q(nc) = -1;
eq    = div(grad(p))+q;
eq(1) = eq(1) + p(1);
p     = -eq.jac{1}\eq.val;
plotCellData(G,p);
```



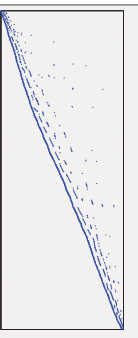
Notice that the D matrix has almost the same sparsity structure as in our first problem, except that the nonzero bands now are curved because the number of cells in each column/row of the grid changes throughout the domain.

Example 4.4.3 *To illustrate the power of the combination of an unstructured grid format and discrete differential operators, we also go through how you can use this technique to*

https://doi.org/10.1017/9781108591416.007 Published online by Cambridge University Press

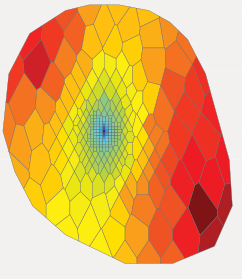
solve the Poisson equation on an unstructured grid. As an example, we use the Voronoi grid generated from the `seamount` *data set shown in Figure 3.11 on page 68. Now comes the important point:* Because the discrete differential operators are defined in terms of the two general matrices C_1 and C_2 that describe the internal connections in the grid, their construction remains *exactly* the same as for the Cartesian grid:

```
load seamount
G = pebi(triangleGrid([x(:) y(:)], delaunay(x,y)));
G = computeGeometry(G);

C  = G.faces.neighbors;
C  = C(all(C ~= 0, 2), :);
nf = size(C,1);
nc = G.cells.num;
D  = sparse([(1:nf)'; (1:nf)'], C, ...
     ones(nf,1)*[-1 1], nf, nc);
grad = @(x) D*x;
```



Here, the directional derivatives do not follow the axial directions and hence D will have a general sparse structure and not the banded structure we saw for the Cartesian grids in the previous example. Moreover, because the cell centers are no longer equidistant points on a uniform mesh, the diagonal entries in the transmissibility matrix will not be the same constant for all cells and hence cannot be scaled out of the discrete system. For historical reasons, MRST only supplies a routine for computing half-transmissibilities defined in (4.51) on page 132. These are defined for all *faces in the grid. Since we have assumed no-flow boundary conditions, we only need to find the half-transmissibilities associated with the* interior *faces, and compute their harmonic average to get the transmissibilities defined in (4.52):*

```
hT = computeTrans(G, struct('perm', ones(nc,1)));
cf = G.cells.faces(:,1);
T  = 1 ./ accumarray(cf, 1 ./ hT, [G.faces.num, 1]);
T  = T(all(C~=0,2),:);

p = initVariablesADI(zeros(nc,1));
q = zeros(nc, 1);  q([135 282 17]) = [-1 .5 .5];
eq    = div(T.*grad(p))+q;
eq(1) = eq(1) + p(1);
p     = -eq.jac{1}\eq.val;
```



You may also notice that we have changed our source terms slightly so that there is now a fluid sink at the center and fluid sources to the northwest and southeast. We will return to a more detailed discussion of the computation of transmissibilities and assembly of discrete linear systems in Section 5.2.

https://doi.org/10.1017/9781108591416.007 Published online by Cambridge University Press

4.4.3 Time-of-Flight and Tracer

The transport equation (4.40) and (4.42) are examples of a more general class of steady-state transport problems,

$$\nabla \cdot (u\vec{v}) = h(\vec{x}, u), \tag{4.64}$$

where $u = \tau$ and $h = \phi + \tau \nabla \cdot \vec{v}$ for time-of-flight and $u = C$ and $h = q_C + C\nabla \cdot \vec{v}$ for the artificial tracer. To discretize the steady transport equation (4.64), we integrate it over a single grid cell Ω_i and use Gauss's divergence theorem to obtain

$$\int_{\partial\Omega_i} u\,\vec{v} \cdot \vec{n}\, ds = \int_{\Omega_i} h\,(\vec{x}, u(\vec{x}))\; d\vec{x}.$$

In Section 4.4.1 we discussed how to discretize the flux over an interface Γ_{ik} between two cells Ω_i and Ω_k for the case that $u \equiv 1$. To be consistent with the notation used above, we will call this flux v_{ik}. If we can define an appropriate value u_{ik} at the interface Γ_{ik}, we can write the flux across the interface as

$$\int_{\Gamma_{ik}} u\,\vec{v} \cdot \vec{n}\, ds = u_{ik} v_{ik}. \tag{4.65}$$

The obvious idea of setting $u_{ik} = \frac{1}{2}(u_i + u_k)$ gives a centered scheme that is unfortunately notoriously unstable. To get a better approximation, we can inspect the direction information propagates in the transport equation. If the flux v_{ik} is positive, the content of u in cell Ω_i is continuously streaming into cell Ω_k, and it is thus natural to choose u_i as the interface value, or vice versa. This gives us the so-called upwind value

$$u_{ik} = \begin{cases} u_i, & \text{if } v_{ik} \geq 0, \\ u_k, & \text{otherwise.} \end{cases} \tag{4.66}$$

Using the upwind value can be thought of as adding extra numerical dispersion to stabilize the resulting scheme so that it does not introduce spurious oscillations; more details about such upwind schemes are given in Chapter 9.

For completeness, let us also write this discretization using the abstract notation defined in the previous section. If we discretize u by a vector $\boldsymbol{u} \in \mathbb{R}^{n_c}$ and h by a vector function $\boldsymbol{h}(\boldsymbol{u}) \in \mathbb{R}^{n_c}$, the transport equation (4.64) can be written in the discrete form

$$\mathtt{div}(\boldsymbol{u}\boldsymbol{v}) = \boldsymbol{h}(\boldsymbol{u}). \tag{4.67}$$

We also substitute the expression for $\boldsymbol{v}$ from (4.62b) and use (4.66) to define u at each face f. Then, we define for each face $f \in S_f$,

$$(\boldsymbol{u}\boldsymbol{v})[f] = \boldsymbol{u}^{uw}[f]\,\boldsymbol{T}[f]\,\mathtt{grad}(\boldsymbol{p})[f], \tag{4.68}$$

where

$$\boldsymbol{u}^{uw}[f] = \begin{cases} \boldsymbol{u}[C_1(f)]), & \text{if } \mathtt{grad}(\boldsymbol{p})[f] > 0, \\ \boldsymbol{u}[C_2(f)]), & \text{otherwise.} \end{cases} \tag{4.69}$$

https://doi.org/10.1017/9781108591416.007 Published online by Cambridge University Press

Time-of-flight and tracer distributions can of course also be computed if we trace individual streamlines by solving the ordinary differential equation (4.36). The most commonly used method for tracing streamlines on hexahedral grids is a semi-analytical tracing algorithm introduced by Pollock [253], which uses analytical expressions of the streamline paths inside each cell, based on the assumption that the velocity field is piecewise linear locally. Pollock's method is only correct for regular grids but is still used for highly skewed and irregular grids. Other approaches for tracing on unstructured grids and the associated accuracy are discussed in [74, 256, 150, 205, 122, 204, 166]. On unstructured polygonal grids, tracing of streamlines becomes significantly more involved. Because the general philosophy of MRST is that solvers should work independent of grid type – so that you can seamlessly switch from structured to fully unstructured, polygonal grids – we prefer to use finite-volume methods rather than streamline tracing to compute time-of-flight and tracer distributions. The disadvantages of this choice are that finite-volume methods have lower pointwise accuracy and that averaged time-of-flight value have a less intuitive interpretation. We will return to a more in-depth discussion of how these quantities can be used to investigate timelines and volumetric connections in the reservoir in Chapter 13.

Computer exercises

4.4.1 Compare the discrete differentiation operators for selected grids from Chapter 3, e.g., Exercises 3.1.1–3.1.3, 3.2.6, and 3.2.7. Can you explain the differences?

4.4.2 Go back to Example 4.4.2 and extend the solver to also compute time-of-flight by using the discrete operators defined in the previous exercise in this chapter.

4.4.3 Extend the solver from Example 4.4.3 to compute one tracer associated with each source term. Together, the two tracers should define a partition of unity for the reservoir volume. How would you visualize it?

https://doi.org/10.1017/9781108591416.007 Published online by Cambridge University Press