Upscaling Immiscible Two-Phase Flows in an Adaptive Frame

Thesis by

Theofilos Strinopoulos

In Partial Fulfillment of the Requirements

for the Degree of

Doctor of Philosophy

California Institute of Technology

Pasadena, California

2005

(Defended November 30, 2005)

To Thanos.

I could only run this far because you tied my shoe laces.

Acknowledgements

I want to thank my advisor Professor Thomas Hou for the opportunity to work with him. This thesis acquires more meaning within the context of the rest of his research because all his projects revolve around the same ideas and shed light to them from different angles. I admire his vision of complete understanding that he strives for and it has been a pleasure to see this understanding slowly be revealed. I want to thank him also for his guidance during these years and his patience with my many shortcomings. He constantly raised the standards and pushed me towards excellence.

Professor Yalchin Efendiev contributed his mathematical insights and experience with the two-phase flow equations. His energy drove this thesis to completion. He oversaw large parts of the my efforts and was always available to discuss and answer my questions. In the days I spent in Texas with him, we resolved some key theoretical points of this work. I enjoyed working with him and am glad that he and his collaborators will continue this work.

Professor Ruo Li helped me with the design of the moving mesh. Professor Danping Yang helped me understand the projection operator in Andy Westhead's thesis. I enjoyed speaking the universal language of mathematics with him.

I want to thank the members of my committee Professors Oscar Bruno, Niles Pierce and Dale Pullin for their suggestions.

It has been a pleasure to share this adventure with the people in the Applied Mathematics Department. I want to thank my classmates Gaby Stredie, Andy Westhead, Lei Zhang, Wuan Luo, Xinwei Yu, Marco Latini, Stephane Lintner, David Hoch for their company, our discussions. Danny Petrasek and Alexei Novikov are another two people whose company and advice I seeked often. I thank Sheila and Chad for keeping the gods of bureaucracy and computers at bay.

My first advisor, Professor Thanos Tzavaras inspired me to undertake this journey. I admire his quality as a person, which I prefer not to divide into words and his love for applied mathematics.

What gave meaning to this journey and the most important thing I take from it are the people I met. My co-protagonists in this colorful drama of success and failure, loneliness and companionship, competition and comraderie, miscommunication and genuine exchange. The curtain has fallen and some of their names appear: Orestis Raptis, Savvas Koudounas, Manos Fitrakis, George Mathaiou, Endy Min, Roger Revilla, Serena Guarnaschelli, Joseph Tsidulko, Sarah Farivar, Dave Michalak, Lavanya Reddy.

Abstract

We derive the two-scale limit of a linear or nonlinear saturation equation with a flow-based coordinate transformation. This transformation consists of the pressure and the streamfunction. In this framework the saturation equation is decoupled to a family of one-dimensional nonconservative transport equations along streamlines. This simplifies the derivation of the two-scale limit. Moreover it allows us to obtain the convergence independent of the assumptions of periodicity and scale separation. We provide a rigorous estimate on the convergence rate. We combine the two-scale limit with Tartar's method to complete the homogenization.

To design an efficient numerical method, we use an averaging approach across the streamlines on the two-scale limit equations. The resulting numerical method for the saturation has all the advantages in terms of adaptivity that methods have. We couple it with a moving mesh along the streamlines to resolve the shock more efficiently. We use the multiscale finite element method to upscale the pressure equation because it gives access to the fine scale velocity, which enters in the saturation equation, through the basis functions. We propose to solve the pressure equation in the coordinate frame of the initial pressure and saturation, which is similar to the modified multiscale finite element method.

We test our numerical method in realistic permeability fields, such as the Tenth SPE Comparative Solution Project permeabilities, for accuracy and computational cost.

Contents

List of Figures

List of Tables

Notation

Bold letters indicate vectors or matrices, normal letters indicate scalars. For the quantities that define the coordinate transformations a capital letter denotes a function and a lowercase letter denotes a spatial variable. For $P, p, \Psi, \psi, \mathbf{v}$ a zero subscript indicates the quantity at $t = 0$. A $\tilde{\cdot}$ denotes a quantity upscaled along a streamline and a $\bar{\cdot}$ a quantity upscaled both along and across the streamlines.

S: Saturation

η: Entropy function for S

q: Entropy flux function for S

Π: Physical pressure

P: Pressure as a function, defined as $P = 1 - \Pi$

p: Pressure as a spatial variable

Ψ: Streamfunction as a function

ψ: Streamfunctions as a spatial variable

$\nabla^{\perp}\cdot$: The operator that maps the velocity vector to the vorticity, $(-\partial_y, \partial_x)$

ξ: Fast spatial variable in the pressure direction, defined as $\xi = \frac{p}{\epsilon}$

ζ: Fast spatial variable in the streamfunction direction, defined as $\zeta = \frac{\psi}{\epsilon}$

$\mathbf{v} = (u, v)$: Velocity in the Cartesian frame

v_0: Velocity in the pressure-streamline frame

$\mathbf{n}$: Unit normal vector

$\mathbf{t}$: Unit tangent vector

i, j: indices for discrete quantities

h: mesh size

$\|f(x,y)\|_n = \left(\int f(x,y)^n dxdy\right)^{1/n}$

ϕ: Test function

Chapter 1

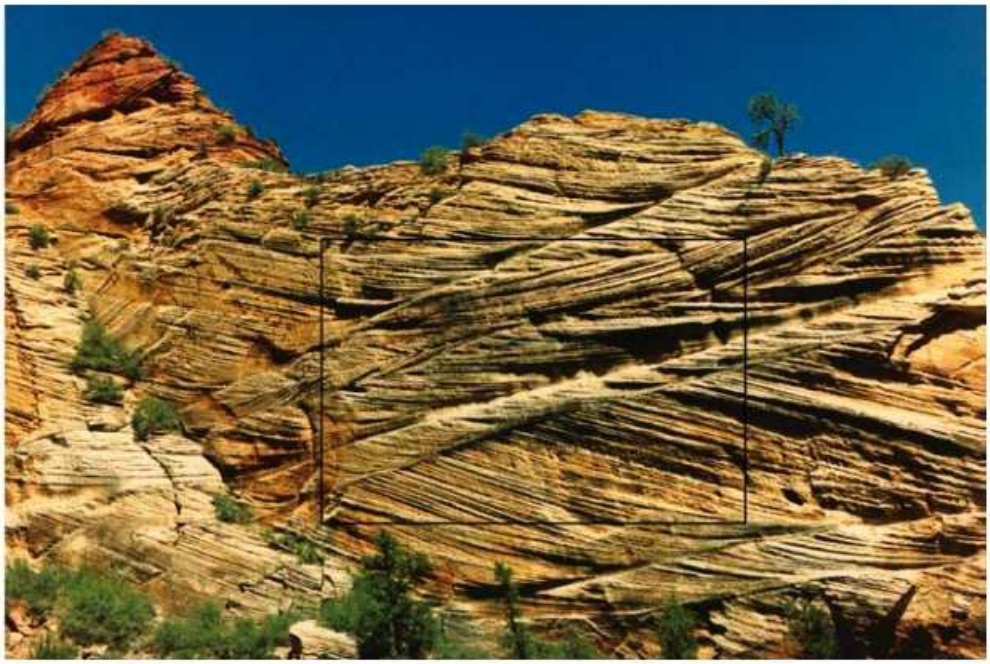
Introduction

Making mistakes is part of academic freedom.

In nature there is a multitude of phenomena whose fundamental physics takes place on a small length scale but we observe them on a larger length scale. We are not interested in water particle collisions but in the motion of rivers; we are not interested in the transfer of electrons in a metal lattice but in the flow of electricity in a wire. Using fundamental physics such as Newton's and Coulomb's laws on the nanoscale we can derive the Navier-Stokes equations and Ohm's law that are valid for larger scales. We will refer to the process of deriving an equation with quantities defined on a large scale from the equation for small length scales as upscaling. In these two examples the physical concepts and the type of the equations differ much between the small and large scale descriptions. This process is far from trivial even in the case when the physical description remains the same. For weather prediction, the physical laws on the scale of meters are the Navier Stokes equation with heat transport, and it is next to impossible to predict the average temperature over a length scale of kilometers.

Flow in porous media is another phenomenon that falls in this category. A porous medium is a solid material that is permeated with miniscule pores that allow the transport of a fluid through it like a sponge. An example of a porous medium in nature is a subsurface rock formation as in figure (1.0.1). Flows in porous media with practical significance arise in enhanced oil recovery and contaminant transport in groundwater. During enhanced oil recovery, a substance such as water or carbon dioxide is pumped through a well to the surrounding porous medium that is saturated with oil. The water or carbon dioxide displaces the oil into nearby wells from

FIGURE 1.0.1. A porous rock



which it can be collected. During contaminant transport in ground water a polluting substance contaminates a water reservoir. They can be modeled by the equations of incompressible immiscible two-phase flow

$$\begin{aligned} \nabla\lambda(S)K(x,y)\nabla P &= 0 \\ \mathbf{v} &= \lambda(S)K\nabla P \\ S_t + \mathbf{v}\cdot\nabla f(S) &= 0, \end{aligned}$$

where P is the pressure, S is the saturation, which is the volumetric ratio of oil or contaminated water to total liquid, and K is the permeability. Note that the pressure equation is coupled to the saturation equation in a nonlinear fashion.

Experiments performed in laboratories can characterize the permeability and other properties of porous rocks on the micron scale, and well logs provide data on the scale of centimeters. Seismic imaging is used to extrapolate the information between wells. Both enhanced oil recovery and contaminant transport involve transport over a length scale of kilometers. The large difference between the length scale on which we know the properties of the rock and the scale over which it is transported, the length scale

of the average flow that we are interested in, makes the cost of direct simulation prohibitive. Even if it were feasible, a direct simulation would waste computational resources because it would provide a much more detailed description than is needed to assess whether enough oil would be recovered or the water supply can be used.

There have been a number of efforts to upscale flow in porous media. Among the general categories of upscaling methods are averaging, percolation theory, homogenization, and hybrid approaches. In an averaging method we derive a cascade of equations, with mean flow and higher-order fluctuations, and we obtain a description on a coarse level by truncating this cascade to some order. In homogenization we examine a sequence of problems with decreasing pore size but constant pore volume and consider the limiting process to describe the upscaled flow. Averaging and homogenization both assume a more or less homogeneous pore structure. This assumption fails in the presence of one connected pore that runs through a large part of the domain, which is referred to as a fast channel, or more generally in the presence of structures with long-range correlation. Such structures will be missed by an upscaling procedure that uses only local information. Such structures arise naturally in porous media near fault lines and cracks in the rock as in figure (1.0.1). Incorporating them accurately in an upscaled model is crucial because most of the transport occurs through them. Percolation theory is based on the opposite assumption as homogenization and averaging, that transport occurs only through a network of such fast channels and provides models where the connectivity is important rather than the local description.

Our approach is based on homogenization and averaging, but in a setting that will make up for these shortcomings. We will combine these two approaches with an adaptive coordinate transformation defined by the pressure P and the streamfunction Ψ. The transformation will detect and account for fast channels and resolve the interface region between oil and water where the largest changes in the flow profile occur. We will use homogenization and averaging to treat the rest of the flow regions where the pore structure is indeed more or less homogeneous. Many upscaling methods that have been proposed in the literature are ad hoc. Instead we will try to derive

a rigorous estimate of the upscaling error and provide a solid mathematical foundation for our upscaling methodology. We will also test it for benchmark permeability models such as the Tenth SPE Comparative Solution Project [**11**] and the synthetic channelized fields considered in [**10**] both for accuracy and computational cost.

The coordinate transformation that we use was inspired by the successes of the time of flight method when applied to the saturation equation. The streamfunction Ψ was first introduced by Muskat [**38**] in the context of two-phase flows. It was first used by Higgins and Leighton [**26**] in the context of a numerical method to define streamtubes through which transport occurs. They mapped the transport equation to the streamtubes and used the fact that it becomes one-dimensional to solve it. Thiele [**50**] followed the same approach, but accounted for the changes in mobility by periodically recalculating the width of the streamtubes so his method bears some resemblance to ours. Streamline methods were introduced by Shafer [**45**] who tracked particles through the domain. Pollock [**44**] used the exact solution of the particle trajectories under the assumption that the velocity field is piecewise linear. Datta Gupta and King [**15**] introduced a new coordinate along the streamlines, the τ, in which the saturation equation has a particularly simple form: $S_t + f(S)_\tau = 0$. Blunt, Lui, and Thiele extended the streamline method to include diffusion and gravity effects, which lead to the modern streamline methods. Streamline and streamtube methods automatically resolve fast channels by virtue of discretizing the flow in Ψ. More importantly when we use Ψ the saturation equation becomes one-dimensional. Complementing our coordinate transformation with the pressure P instead of τ results in an orthogonal transformation. We also note that in time of flight methods it is difficult to include cappilary pressure effects whereas with our formulation it is straightforward. With P instead of τ the saturation equation retains its multiscale structure so that upscaling is meaningful.

Our homogenization approach is based on Nguytseng's theory of two-scale convergence. Nguytseng [**41**] proposed using oscillatory test functions $\phi(x, y, \frac{x}{\epsilon}, \frac{y}{\epsilon})$ to define the homogenization limit. Such a limit would be an average over all length scales leaving the fast scales $\frac{x}{\epsilon}, \frac{y}{\epsilon}$ intact. If we try to derive the two-scale limit in a periodic domain we stumble upon the fact that the cell problem $\mathbf{v} \cdot \nabla_{\left(\frac{x}{\epsilon}, \frac{y}{\epsilon}\right)} S = 0$ has

no unique solution. This equation for the cell problem means that the saturation to lowest order is constant along streamlines. To restrict the degrees of freedom contained in a general two-scale expansion, E [**19**] incorporated this piece of information to the expansion by using a test function $\phi(x, y, \frac{x}{\epsilon}, \frac{y}{\epsilon})$ with $\mathbf{v} \cdot \nabla_{(\frac{x}{\epsilon}, \frac{y}{\epsilon})} \phi = 0$. E arrives at a unique two-scale limit, but the equations by which the two-scale limit is defined don't offer any insight into its structure. Moreover, there is no clear way to design an efficient numerical method based on them. Westhead [**53**] incorporates the fact that the saturation to lowest order is constant along the fast streamlines by introducing a projection operator onto the average along the fast streamlines and also obtains a unique two-scale limit. He expands the saturation equation into an average and fluctuations and uses the projection operator to close the equation for the fluctuations. The resulting equations are much more revealing about the structure of the two-scale limit, and a numerical method is proposed.

The approach of Westhead assumes a two-scale structure for the velocity field in the Cartesian variables. We will assume a fast variable in the pressure-streamline frame. The upscaling philosophy in this thesis for upscaling either the pressure or the saturation equation is that the geometry of the flow should be reflected in the coordinate system that we use. Also we will not apply the projection only on the fluctuations as Westhead did but will restrict the test functions as E. The implementation of this idea is the main result of this thesis. The idea to use a flow-based coordinate system in the context of homogenization of the incompressible Euler equations appeared first in McLaughlin, Papanicolaou, Pironneau [**40**]; however they used a slightly different multiscale expansion than we did and did not arrive at a unique homogenization limit.

On a theoretical level the saturation equation in P, Ψ is one-dimensional which allows us to obtain a convergence rate to the two-scale limit that is independent of the assumptions of periodicity and scale separation. We can also carry out the homogenization farther than the two-scale limit using Tartar's method. On a practical level the homogenized equations that we arrive at are much simplified compared to those of Westhead. In addition consider how the fast flow channels are upscaled with a Cartesian and a pressure-streamline method. In the Cartesian frame they are thin

and they are grouped together with slow channels resulting in a large upscaling error. In the pressure-streamline frame they are wider because the transformation focuses the computational points in the fast regions and they are grouped separately from the slow channels. This leads to a more accurate upscaling method. We apply our upscaling philosophy to the pressure equation and complete our upscaling method for the two-phase flow equations.

In the second chapter we describe the physical phenomena that occur during flows in porous media, present their equations, and demonstrate the difficulties that arise in modeling. We select the simplified model of side-to-side incompressible flow with no capillary effects, which retains the essential difficulties in upscaling, the presence of small scales that affect the large scale flow in a significant way. While describing a naive approach to solve numerically the two-phase flow equations we introduce concepts involved in designing a numerical method for the two-phase flows that will be used in later chapters to design an accurate upscaling method. Finally we show a few numerical results and highlight the essential features in the solution such the shock at the interface and the presence of long fingers.

In the third chapter we introduce the coordinate transformation to the frame defined by the pressure and the streamfunction. Our upscaling method that is described in the next chapters will be based on this coordinate system so we describe its properties in detail. In a pressure-streamline coordinate frame the two-dimensional transport equation is decoupled into a family of one-dimensional equations and the grid is focused in the high flow regions. We show how these two properties can be used together with a moving mesh to design a method that is easy to implement and superior to the method in the Cartesian frame, described in the first chapter. In numerical experiments with realistic permeability fields we observe that the numerical method in the pressure-streamline frame converges faster, that the error constant is smaller. This concludes the part where the discretization grids resolve the fine scale of the permeability and leads us to chapters three and four where the discretizations do not resolve the small scales.

In the fourth chapter we upscale the hyperbolic equation of the two-phase flow equations. For that we restrict ourselves to one-phase flow. When we look at the

velocity field in the pressure-streamline coordinates the fast channels are magnified and the variation of the velocity field over the coarse scale is smaller. The flow is more or less uniform in the coarse cells of a coordinate system that follows the flow. We obtain the homogenization of the equation, using two-scale convergence that had been obtained in the Cartesian scheme in the nineties. The two results are different because they are valid under assumptions of two different multiscale structures. Yet we compare them and conclude that the approach in the pressure-srteamline frame leads to simpler equations that are easier to solve. We show weak convergence and rigorously estimate the upscaling error of the two-scale limit without using the assumption of periodicity and scale separation. Then we average out the fast dependence of the two-scale limit across the streamlines using Tartar's method to obtain a fully homogenized equation. It is not clear how to design an efficient method using that equation, so we use an averaging approach to remove the fast dependence of the two-scale limit across the fast streamlines following Efendiev and Durlofsky [**20, 21**]. The advantages of the resulting method are demonstrated on realistic permeability fields in terms of accuracy and time of computation.

In the fifth chapter we apply the same upscaling philosophy to the elliptic equation. We use the multiscale finite element method that is based on the homogenization theory for elliptic operators. Before we apply the method we transform to the pressure-streamline coordinates to deal with the long-range structures of the permeability field. Our approach has all the advantages of an improved version of the multiscale finite element method, called the modified multiscale finite element method, and also treats fast channels more effectively. This leads to very accurate upscaling as we demonstrate in numerical experiments. Finally we combine the two approaches and present numerical results for the full two-phase flow.

Part 1

Resolved Scheme

In the first part we introduce the equations and the physical phenomena that they describe. We examine the properties of the pressure-streamline frame in terms of adaptivity. We assume that all the quantities are smooth when we discretize them. That is, if we denote by ϵ the length scale on which the quantities fluctuate, then we assume that our grid is such that $\epsilon > h$ and it resolves all scales.

Chapter 2

The Porous Media Equations in the Cartesian Frame

2.1. Derivation of the Physical Model

We will start by introducing the phenomena that take place in a porous medium and deriving a quantitative description for them. A porous medium is a solid medium that is permeated by miniscule pores that allow the transport of a gas or a fluid. That gas or fluid is often referred to as "phase". We will restrict ourselves to immiscible flows, flows where the two phases are separated by a distinct boundary and don't mix. We will also consider phases that do not react with each other. In general, the motion of any fluid is described by the Navier-Stokes equations. For a flow in a small pipe the Reynolds number, which quantifies the relative importance of convection of momentum to viscous diffusion of momentum, is $\frac{VL}{\nu}$, where V is the velocity, L is the characteristic length scale, the diameter of the pipe in this case, and ν is the kinematic viscosity. In our case all three quantities are such that the Reynolds number is very small and in this limit the Navier-Stokes equations reduce to the linear Stokes equations.

The pores are on a much smaller length scale than the length scale of the average transport, so there have been early on attempts to derive a physical law on a larger length scale. In 1856, Darcy discovered empirically a law, which now bears his name, and describes the transport of one fluid in a porous medium. Darcy's law can be

written as

$$\mathbf{v} = -\frac{K}{\mu}(\nabla\Pi - \rho g \nabla D), \tag{2.1.1}$$

where $\mathbf{v}$ is the velocity of the fluid, K is the permeability of the medium, μ is the viscosity, Π is the pressure, ρ is the density of the fluid, g is the constant of gravity, and D is depth. Darcy's law essentially states that the velocity of the fluid is proportional to the gradient of pressure. The permeability K is an empirical constant that depends only on the geometry of the porous medium and describes how fast the fluid moves through the region for a given pressure gradient. The same law has been recently derived from the Stokes equations using homogenization (for the details we refer the interested reader to [**27**]). Darcy's law is complemented by an equation for the conservation of mass of the fluid

$$\nabla \cdot (\rho \mathbf{v}) + q = \frac{\partial(\phi\rho)}{\partial t}, \tag{2.1.2}$$

where q denotes sources or sinks in the domain, and ϕ is the ratio of the pore volume available to the fluid to the total volume, referred to as porosity. Porosity describes how much volume is available to the fluid and does not depend on the geometry of the pores, while permeability describes how the geometry and connectivity of the pores affect the mean flow. We can consider the porosity to be the density of the rock. Passive transport of contaminant through the water of porous medium can be described by (2.1.1), (2.1.2). It is referred to as one-phase flow.

For flows of two different fluids in a porous medium, Muskat [**39**] first postulated that Darcy's law holds for each fluid separately

$$\begin{aligned} \mathbf{v}_w &= -\tfrac{K_w}{\mu_w}(\nabla\Pi_w - \rho_w g \nabla D) \\ \mathbf{v}_n &= -\tfrac{K_n}{\mu_n}(\nabla\Pi_n - \rho_n g \nabla D). \end{aligned} \tag{2.1.3}$$

The two different fluids have been denoted with subscripts n and w for "non-wetting" and "wetting". This refers to the fact that the walls of the porous medium show preference in being covered by one of the two fluids. Which fluid covers the walls of the porous medium determines the contact angle of the interface of the two fluids with the wall and the capillary pressure. Usually water is more wetting than oil and

a gas would be the least wetting phase of the three. The difference of the pressures of the two phases is equal to the capillary pressure

$$\Pi_n - \Pi_w = \Pi_c. \tag{2.1.4}$$

Darcy's law for one fluid (2.1.1) is much simpler to work with than for two fluids, as in (2.1.3), because in (2.1.1) the permeability K depends only on the porous medium whereas in (2.1.3) K_w is the permeability of the porous medium combined with the non-wetting phase and obviously depends on the distribution of the non-wetting phase, likewise for K_n. We refer to K_w, K_n as the relative permeability. Our goal is to derive equations for the flow of two fluids where the influence of the pore structure and the influence of one fluid on the other are separated and their nature is elaborated. Then the resulting equations would be as simple as the equations for one fluid in a certain sense.

Experiments have shown that the relative permeabilities depend only on the ratio of the concentration of the fluids that are involved and are more or less independent of their type of fluids, their velocities and pressures, or the past flow profiles. This means physically that the flow reaches its steady state locally. For a discussion of cases when this assumption fails and the flow is not at equilibrium locally we refer to Barenblatt [**3**]. To utilize this experimental observation we will introduce the concepts of saturation and phase mobilities.

We will denote by $S_w(x, y)$ the ratio of pore volume occupied by the wetting phase to the total pore volume and refer to it as the saturation of the wetting phase, likewise for S_n. By definition we will have

$$S_w + S_n = 1. \tag{2.1.5}$$

The phase mobility λ_w is defined as $\lambda_w = \frac{K_w}{K\mu_w}$, likewise for the non-wetting phase. We write Darcy's law in terms of the phase mobilities

$$\begin{aligned} \mathbf{v}_w &= -\lambda_w K(\nabla\Pi_w - \rho_w g\nabla D) \\ \mathbf{v}_n &= -\lambda_n K(\nabla\Pi_n - \rho_n g\nabla D). \end{aligned} \tag{2.1.6}$$

In this framework the experimental observation that relative permeability only depends on the ratio of the concentrations can be translated as

$$\lambda_w = \lambda_w(S_w) \text{ and } \lambda_n = \lambda_n(1 - S_w).$$

The conservation of mass (2.1.2) can be written in terms of the saturations and velocities of each phase as

$$\begin{aligned} \nabla \cdot (\rho_w \mathbf{v}_w) + q_w &= \tfrac{\partial(\phi \rho_w S_w)}{\partial t} \\ \nabla \cdot (\rho_n \mathbf{v}_n) + q_n &= \tfrac{\partial(\phi \rho_n S_n)}{\partial t}. \end{aligned} \tag{2.1.7}$$

Equations (2.1.7), (2.1.3) are general and can apply to flows with any number of phases. We have included the effects of compressibility of the phases and the porous medium and of capillary pressure.

2.1.1. Pressure Equation. To derive the pressure equation analogous to the pressure equation for one-phase flow we first define an average pressure Π as

$$\Pi = \frac{\Pi_w + \Pi_n}{2}.$$

We can write the pressure of the wetting and non-wetting phases in term of the average pressure as

$$\begin{aligned} \Pi_w &= \Pi - \tfrac{\Pi_c}{2} \\ \Pi_n &= \Pi + \tfrac{\Pi_c}{2}. \end{aligned} \tag{2.1.8}$$

We combine (2.1.7), (2.1.3) to obtain

$$\begin{aligned} \nabla \cdot (\rho_w (-\lambda_w K (\nabla \Pi_w - \rho_w g \nabla D))) + q_w &= \tfrac{\partial(\phi \rho_w S_w)}{\partial t} \\ \nabla \cdot (\rho_n (-\lambda_n K (\nabla \Pi_n - \rho_n g \nabla D))) + q_n &= \tfrac{\partial(\phi \rho_n S_n)}{\partial t}. \end{aligned} \tag{2.1.9}$$

Quantities ϕ, ρ_w, ρ_n depend on the pressure since the phases and the medium were assumed to be compressible. We define compressibility as the relative change in volume divided by the change in pressure required to achieve it. Mathematically, we

let

$$\begin{aligned} C_\phi &= \frac{1}{\phi}\frac{d\phi}{d\Pi} \\ C_{_n} &= \frac{1}{\rho_n}\frac{d\rho_n}{d\Pi_n} \\ C_{_w} &= \frac{1}{\rho_w}\frac{d\rho_w}{d\Pi_w}. \end{aligned}$$

We make this dependence of the right hand side of (2.1.9) on the pressures explicit by expanding the time derivatives and writing (2.1.9) in terms of the compressibilities

$$\begin{aligned} (2.1.10) \qquad \nabla\cdot\left(\rho_w\left(-\lambda_w K\left(\nabla\Pi_w-\rho_w g\nabla D\right)\right)\right)+q_w &= \rho_w S_w\phi C_\phi\frac{\partial\Pi}{\partial t} \\ &\quad +\phi S_w\rho_w C_{_w}\frac{\partial\Pi_w}{\partial t}+\phi\rho_w\frac{\partial S_w}{\partial t} \\ (2.1.11) \qquad \nabla\cdot\left(\rho_n\left(-\lambda_n K\left(\nabla\Pi_n-\rho_n g\nabla D\right)\right)\right)+q_n &= \rho_n S_n\phi C_\phi\frac{\partial P}{\partial t} \\ &\quad +\phi S_n\rho_n C_{_n}\frac{\partial\Pi_n}{\partial t}+\phi\rho_n\frac{\partial S_n}{\partial t}. \end{aligned}$$

Now we divide (2.1.10) by ρ_w and (2.1.11) by ρ_n and add the results to eliminate the time derivatives of S_w, S_n. We use equations (2.1.5), (2.1.8) to obtain an equation for Π

$$\begin{aligned} &\left(-\frac{1}{\rho_w}\nabla\cdot(\rho_w\lambda_w K\nabla)-\frac{1}{\rho_n}\nabla\cdot(\rho_n\lambda_n K\nabla)\right)\Pi \\ &+\left(\frac{1}{2\rho_w}\nabla\cdot(\rho_w\lambda_w K\nabla)-\frac{1}{2\rho_n}\nabla\cdot(\rho_n\lambda_n K\nabla)\right)\Pi_c \\ &+\left(\frac{1}{\rho_w}\nabla\cdot(\rho_w^2 Kg\nabla)+\frac{1}{\rho_n}\nabla\cdot(\rho_n^2 Kg\nabla)\right)D+Q \\ &\quad=\phi(C_\phi+S_wC_w+S_nC_n)\frac{\partial\Pi}{\partial t}+\frac{\phi}{2}(-S_wC_w+S_nC_n)\frac{\partial\Pi_c}{\partial t}. \end{aligned}$$

We have denoted the sources and sinks by Q. The resulting equation is parabolic in Π or Π_c.

We would like to make some simplifications in this equation, while still keeping the terms that describe the essential physics and that are the most difficult to model numerically. In practice the capillary pressure is much smaller than the average pressure so we can neglect it. Furthermore we can treat the gravity term as external

forcing in the same way that Q will be treated so we will refer to them as $\nabla \cdot G_p$. The equation becomes

$$\left(-\frac{1}{\rho_w}\nabla\cdot(\rho_w\lambda_w K\nabla) - \frac{1}{\rho_n}\nabla\cdot(\rho_n\lambda_n K\nabla)\right)\Pi + \nabla\cdot G_p + Q = \phi(C_\phi + S_w C_w + S_n C_n)\frac{\partial \Pi}{\partial t}.$$

We regard the quantity $C_\phi + S_w C_w + S_n C_n$ as the total compressibility. Then the above equation is analogous to the corresponding equation for one-phase flow with a time-independent porosity, which arises from the combination of (2.1.1), (2.1.2)

$$\frac{1}{\rho}\nabla\cdot(\rho\lambda K\nabla)\Pi = \phi C\frac{\partial \Pi}{\partial t}.$$

Our final assumption is that the medium and the phases are incompressible. This assumption is justified in the case of fluids moving through the rock. When a gas is used to displace the oil the effects of compressibility cannot be ignored; however the complications that they create can be treated in a similar way as in the incompressible case. Our final equation for the pressure is

$$(-\nabla\cdot(\lambda_w K\nabla) - \nabla\cdot(\lambda_n K\nabla))\,\Pi + \nabla\cdot G_p + Q = 0. \tag{2.1.12}$$

Based on the analogy with one-phase flow we can define a total velocity $\mathbf{v}$ to be

$$\mathbf{v} = (\lambda_w + \lambda_n)K\nabla\Pi + G_p = \mathbf{v}_n + \mathbf{v}_w. \tag{2.1.13}$$

In this work we will neglect the gravity and source terms for simplicity.

2.1.2. Saturation Equation. Following the same line of thought we derive an equation for the saturation that involves average quantities rather than quantities specific to a phase. We substitute Darcy's law for two phases (2.1.3) into (2.1.4) in terms of the gradients to obtain

$$\begin{aligned} \mathbf{v}_n + \mathbf{v}_w &= \mathbf{v} \\ -\frac{\mathbf{v}_n}{\lambda_n K} + \rho_n g\nabla D + \frac{\mathbf{v}_n}{\lambda_n K} - \rho_n g\nabla D &= \nabla\Pi_c. \end{aligned}$$

We can solve for the velocities of each phase in terms of the average velocity and the capillary pressure. We have

$$\mathbf{v}_w = \frac{\lambda_w}{\lambda_w + \lambda_n}\mathbf{v} + \frac{\lambda_w}{\lambda_w + \lambda_n}\lambda_n K \nabla \Pi_c + G_s.$$

We have lumped all the terms involving gravity into G_s. We will make the assumption that the capillary pressure depends only on S_w and write the velocity of the wetting phase as

$$\mathbf{v}_w = f_w \mathbf{v} + h_w \nabla S_w + G_s,$$

where

$$\begin{aligned} f_w &= \frac{\lambda_w}{\lambda_w + \lambda_n} \\ h_w &= \frac{\lambda_w}{\lambda_w + \lambda_n}\lambda_n K \frac{d\Pi_c}{dS_w}. \end{aligned}$$

With this expression for the velocity the equation for the saturation of the wetting phase can be written as

$$\nabla \cdot (\rho_w f_w \mathbf{v}) + \nabla \cdot (\rho_w h_w \nabla S_w) + \nabla \cdot (\rho_w G_s) + q_w = \frac{\partial(\phi \rho_w S_w)}{\partial t}.$$

In the average pressure equation (2.1.12) the sources Q refer to the points of injection of water and collection of oil. Away from these points (2.1.12) shows that the average velocity $\mathbf{v}$ is incompressible. Then we see that the saturation equation is a parabolic equation

$$\mathbf{v} \cdot \nabla(\rho_w f_w(S_w)) + \nabla \cdot (\rho_w h_w \nabla S_w) + \nabla \cdot (\rho_w G_s) + q_w = \frac{\partial(\phi \rho_w S_w)}{\partial t}$$

that describes convection by the average velocity $\mathbf{v}$ and a nonlinear flux function $f(S_w)$ in the presence of diffusion due to capillary forces and external forcing due to gravity. Mathematically, the significance of the diffusion due to capillary pressure is that it selects a physically relevant weak solution for the nonlinear conservation law.

In practice at the length scales that geological data are available, diffusion due to capillary forces is negligible. Moreover modeling and upscaling become more challenging when the transport is convection dominated rather than diffusion dominated. We simplify the saturation equation by omitting the forcing terms due to gravity and

the capillary forces that would add unnecessary complications without illustrating better any of the concepts to be presented. We will also assume that the fluids are incompressible. In the case of oil and water this is a valid assumption; moreover the transport of a compressible fluid must be dealt with in a manner that is consistent with the incompressible case. Finally we assume that the porosity is constant throughout the medium. With these assumptions the saturation equation reduces to

$$\mathbf{v} \cdot \nabla(f_w(S_w) = \frac{\partial S_w}{\partial t}. \tag{2.1.14}$$

This equation was discovered by Buckley and Leverett [**8**] and bears their names. Given S_w we can easily compute S_n by (2.1.5). Equations (2.1.4), (2.1.14) have the desirable property that the influence of the porous medium and the interaction of the two phases have been separated and encoded into K and f_w,λ, a form that is intuitive and can be quantified easily by experimental measurements. The most widely adopted model for λ_w and λ_n is

$$\begin{aligned} \lambda_w(S_w) &= S_w^2 \\ \lambda_n(S_w) &= m(1-S_w)^2, \end{aligned}$$

where m is the mobility ratio. With these constitutive relations we have

$$\begin{aligned} \lambda_w(S_w) + \lambda_w(S_w) &= \lambda(S_w) = S_w^2 + m(1-S_w)^2 \\ \frac{\lambda_w}{\lambda_w + \lambda_n} &= f_w(S_w) = \frac{S_w^2}{S_w^2 + m(1-S_w)^2}. \end{aligned} \tag{2.1.15}$$

One-phase flow can be recovered from the equations of two-phase flow by choosing $\lambda(S) = 1$ and a linear flux $f_w(S_w) = S_w$. We note that for two-phase flow the pressure depends dynamically on the saturation through λ whereas in one-phase flow the pressure and saturation equations are decoupled and the pressure is time-independent.

2.1.3. Model Problem. To make the presentation simpler we restrict ourselves to flows in the unit square. There are two types of two-dimensional flows that are widely considered, side-to-side and corner-to-corner. Corner-to-corner flow refers to flow across opposite corners of the unit square and side-to-side refers to flow across opposite faces. If we consider a three-dimensional underground flow from a well

into which water is pumped to a well where oil is collected, corner-to-corner flow corresponds to a horizontal cross-section and side-to-side corresponds to a vertical cross-section. In this work we will mostly consider side-to-side flows for convenience and show how to treat corner-to-corner flows in a similar way. Instead of the physical pressure Π we will substitute for it $\Pi = 1 - P$ and refer to the new variable as pressure P with a slight abuse of terminology. For the rest of this work we will drop the subscripts w.

Then the equations (2.1.12), (2.1.14) of the previous section for pressure P and saturation S in a Cartesian frame are

$$\begin{aligned} \nabla\lambda(S)K\nabla P &= 0 \\ \mathbf{v} &= \lambda(S)K\nabla P \\ \nabla^2\Psi &= -\nabla^\perp\cdot\mathbf{v} \\ S_t + \mathbf{v}\cdot\nabla f(S) &= 0. \end{aligned} \tag{2.1.16}$$

The boundary and initial conditions are

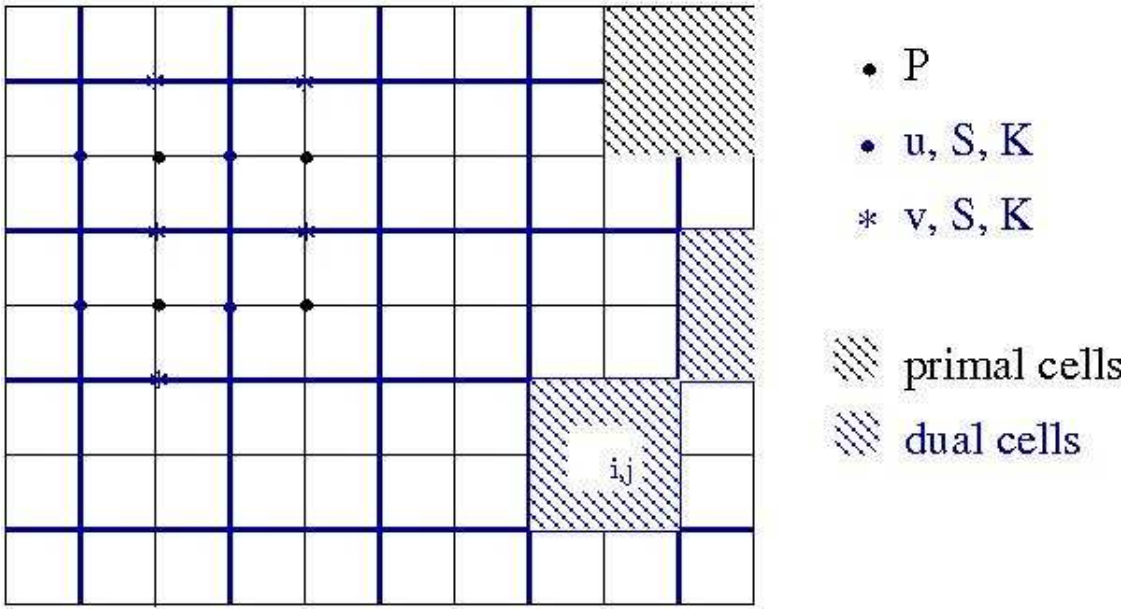
(2.1.17)

$$\begin{array}{lllll} P(x=0)=0 & \partial_y P(y=0)=0 & \Psi(y=0)=0 & \partial_x\Psi(y=0)=0 & S(x=0)=1 \\ P(x=1)=1 & \partial_y P(y=1)=0 & \Psi(y=1)=c & \partial_x\Psi(y=1)=0 & S(t=0)=0. \end{array}$$

We have denoted by $\nabla^\perp\cdot$ the operator that maps the velocity vector to the vorticity, $\nabla^\perp\cdot(v_x, v_y) = -\partial_y v_x + \partial_x v_y$.

Ψ is the streamfunction and is defined to be constant on the streamlines. The streamfunction is not part of the standard set of two-phase flow equations, but is important for our approach so we have added it to (2.1.16). The Dirichlet boundary condition on the pressure designates the $x = 0$ boundary as the inlet, the point where water or contaminant is pumped, and the $x = 1$ boundary as the outlet, the point where it is collected. The Neumann condition for the pressure imposes that there be no outgoing flow at the $y = 0$ and $y = 1$ boundaries. The streamfunction is constant on the integral lines of the velocity field; its level sets follow the flow. The constant c can be chosen arbitrarily. We will chose $c = \int_0^1 u dy$, so that $\Psi_y = u$.

FIGURE 2.2.1. Staggered grid for the pressure



In a finite volume scheme for the pressure equation, the permeability and the velocity are defined on the edges of the dual cells and the pressure is defined in the centers of the dual cells.

2.2. Resolved Scheme

Next we will derive a numerical method for the model problem (2.1.16) and investigate numerically the properties of the solutions. On the way we will illustrate some of the issues involved in the design and how they are tackled. This numerical method will serve as a basis for our upscaled scheme that will be developed in part two. We will also introduce some measures to quantify the flow.

2.2.1. Finite Volume Method for the Pressure equation. For the pressure equation we will employ a finite volume (FV) method. Practically speaking a finite volume method arises from a straightforward finite difference discretisation on a staggered grid. We partition the domain into rectangular cells that define the primal grid, as in figure 2.2.1. Connecting the centers of the cells we obtain the dual grid.

To derive the finite volume discretisation we start from the integral formulation of the pressure

$$\int_{\Omega_{i,j}} \nabla\lambda(S)K\nabla P dA = \int_{\partial\Omega_{i,j}} \mathbf{v}\cdot n dS = 0.$$

This equation holds over all subsets $\Omega_{i,j}$ of our domain, in particular over the cells of the primal grid. If we assume that the velocity is piecewise constant over the edges k of the dual control volumes $\Omega_{i,j}$ and the pressure is piecewise linear in $\Omega_{i,j}$ we arrive at the following discretisation

$$
\begin{aligned}
\sum_{\text{edge } k \text{ of } \Omega_{i,j}} \mathbf{v}_k \cdot n_k &= 0 \\
P_{i+1,j} - P_{i-1,j} &= \frac{\mathbf{v}_k \cdot n_k}{\lambda(S)K} \\
P_{i,j+1} - P_{i,j+1} &= \frac{\mathbf{v}_k \cdot n_k}{\lambda(S)K}.
\end{aligned} \tag{2.2.1}
$$

Note that the velocity $\mathbf{v}_i$ is conservative because it satisfies (2.2.1).

2.2.2. Finite Volume Method for the Saturation Equation. For the time discretisation of the saturation equation in (2.1.16) we use an Implicit Pressure Explicit Saturation Scheme (IMPES). We assume that in the time interval $(t_0, t_0 + \Delta t_P)$ the velocity remains constant and is equal to its value at time t_0. Physically, this is equivalent to assuming that in the time interval $(t_0, t_0 + \Delta t_P)$ we have one-phase flow. For example a first order IMPES method in time would be given by

$$
\begin{aligned}
\nabla\lambda(S^t)K\nabla P^t &= 0 \\
\mathbf{v}^t &= \lambda(S^t)K\nabla P^t \\
\frac{S^{t+\Delta t} - S^t}{\Delta t} - \mathbf{v}^t \cdot \nabla f(S) &= 0.
\end{aligned} \tag{2.2.2}
$$

The nonlinear coupling of the saturation and the pressure equation occurs through the velocity. If we discretize the two-phase flow equations as in (2.2.2) we will obtain an explicit expression of S in terms of quantities at time t_0, in particular in terms of the velocity. The velocity $\mathbf{v}^t$ in turn is coupled to the pressure in an implicit fashion since its formula contains the saturation S^t at time t_0. The scheme will be first-order in time, independently of the rest of the discretization because the velocity is assumed to be piecewise constant in time.

For the spatial discretization of (2.2.2), the cells of the finite volume scheme are the cells of the dual grid. The flux (2.1.15) of the Buckley Leverett equation is nonconvex

and could in general require special treatment in the numerical scheme. For a general nonconvex scalar conservation law, Osher [**42**] discovered a functional relation for the saturation and its flux in terms of the similarity variable $\frac{x}{t}$. Evaluating this relation at $\frac{x}{t} = 0$ corresponds to a stationary frame and gives the flux for the Riemann problem, which is what we need in the construction of the numerical scheme. It turns out to be the same as in the case of convex flux. We defer the rest of the details until section 3.3.3.

It is very crucial that the CFL condition is not violated. In some cases if the CFL condition is violated the saturation does not blow up, but instead we obtain a solution that does not satisfy the entropy condition.

2.3. Some Numerical Results

There are three widely used quantities of interest in two-phase flows that quantify numerical error. We will look at the relative error, defined as $\frac{\|exact-computed\|}{\|exact\|}$. The L_1 norm of the saturation can be interpreted as the total amount of transported fluid in the domain and can be used to investigate whether a method is conservative or not. We can use other L_p norms, but we note here that it is meaningless to consider the maximum norm when the exact solution has a discontinuity as the saturation does. Quantities related more to practical applications are the breakthrough time, which is the time that water first reaches the collection well, the fractional flow rate (ffr), and the pore volumes injected (pvi). The fractional flow rate is defined as

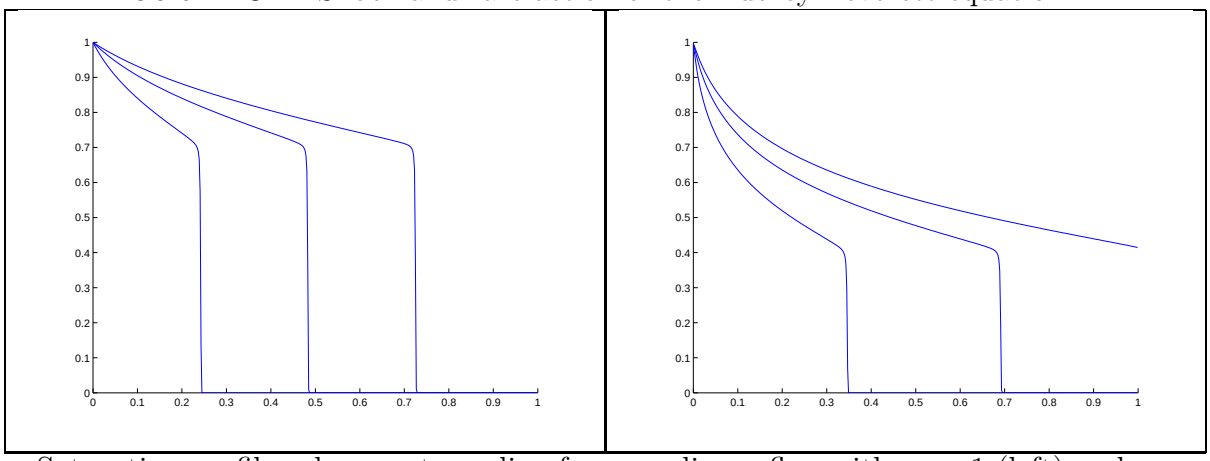
$$\text{ffr}(t) = \frac{\int_0^1 f(S)\mathbf{v}\cdot\hat{\mathbf{x}}dy}{\int_0^1 \mathbf{v}\cdot\hat{\mathbf{x}}dy} \tag{2.3.1}$$

and describes the volumetric proportion of oil to total fluid that is collected at the outlet well, the quantity that is of most immediate use to the petroleum engineer. In a coordinate frame p, ψ that satisfies $\mathbf{v}\cdot\check{\mathbf{x}}dy = d\psi$ the fractional flow rate becomes

$$\text{ffr}(t) = \int_0^1 f(S)d\psi. \tag{2.3.2}$$

The pressure-streamline frame that we will introduce in the next chapter has this property so we will use the second form (2.3.2) in our computations. The fractional flow curve is a plot of the fractional flow rate with time, and its integral is the total

FIGURE 2.3.1. Shock and rarefaction of the Buckey Leverett equation



Saturation profiles along a streamline for a nonlinear flux with $m = 1$ (left) and $m = 5$ (right) at $t = 0.2, 0.4, 0.6$.

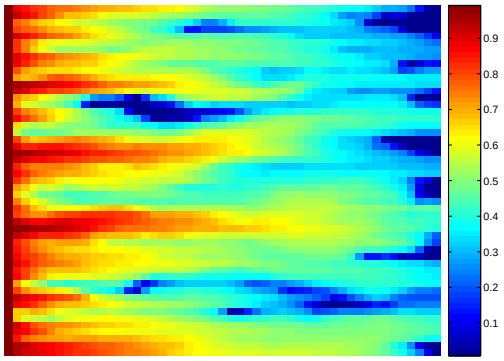
amount of oil produced at the outlet until a certain time. Instead of time, fractional flow curves are often reported as functions of pore volumes injected, defined for unit porosity as

$$pvi = \frac{1}{\|\Omega\|} \int_0^t \int_0^1 \mathbf{v} \cdot \hat{\mathbf{x}} dy,$$

where Ω is the area of the domain. Pvi is the time that is required to inject an amount of fluid equal to the capacity of the oil reservoir, essentially it is a rescaling of time.

We discuss briefly some characteristics of two-phase flows. We show a plot of the saturation profile along a streamline at different times in figure 2.3.1. The saturation develops a shock whose strength diminishes as time passes. The shock is followed by a rarefaction wave. As the mobility increases, the shock strength decreases. In figure 2.3.2 we show a saturation profile in two dimensions computed in Cartesian coordinates with the method described in this chapter. In flows in two and three dimensions when the velocity varies rapidly across the flow lines, the saturation profile develops fingers along the fast channels of the flow. We are interested in resolving the fast channels and upscaling over adaptively selected regions where the flow is more or less uniform. Note that the presence of numerical diffusion smears the shock out. The grid that was used was 60×60, which explains why the resulting saturation profile is smeared. This effect will be much reduced when we use an adaptive coordinate frame, and we will be able to see the fingers more clearly.

FIGURE 2.3.2. Fingering in two phase flows



When the permeability field varies strongly across the flow, the saturation profile develops fingers.

Chapter 3

An Adaptive Framework for Solving the Porous Media Equation

3.1. Context of the Present Work

The most successful attempt to design an adaptive numerical method for the transport equation in oil reservoir simulation and ground water flow is the time of flight method. It was slowly developed by many researchers and expanded to apply to the many physical phenomena that arise in two-phase flows. A detailed review of these efforts can be found in [**4**]. The central idea is to track the streamlines of the flow and solve a one-dimensional transport equation along them

$$S_t + |\mathbf{v}|\frac{d}{ds}f(S) = 0,$$

where s denotes arc length. Datta-Gupta and King [**15**] introduced time of flight, which is central to modern streamline methods. It is defined by

$$\tau = \int_0^s \frac{d\zeta}{|\mathbf{v}(\zeta)|}.$$

If we change variables from s to τ we arrive at an equation whose solution is easy to compute because it does not contain the velocity field

$$S_t + \frac{d}{d\tau}f(S) = 0.$$

Time of flight satisfies the ikonal equation on the plane

$$1 = |\mathbf{v}|\frac{d\tau}{ds} = |\mathbf{v}|\nabla\tau \cdot \frac{\mathbf{v}}{|\mathbf{v}|} \implies \mathbf{v} \cdot \nabla\tau = 1.$$

From this equation it is clear τ is a well-defined function in the plane. (τ, ψ) is not an orthogonal system of coordinates. To see this we can consider the case of shear flow that arises when $K = 1 - y$. Then the velocity field is $\mathbf{v} = (1 - y, 0)$ and the streamlines simply $y = const$. Time of flight is given by $\tau = \frac{x}{|1-y|}$ so its level sets are not perpendicular to those of the streamfunction. The pressure is given by $P = x$ and (p, ψ) is an orthogonal system. This fact holds generally as we will see. Our approach, which will be described in the next section, is based on the coordinate frame p, ψ and has all the advantages in terms of adaptivity as the streamline method. In addition it is easier and more meaningful to try to upscale in p, ψ coordinates compared to (τ, ψ) as will become apparent in part two.

3.2. The Pressure-Streamline Frame

3.2.1. Coordinate Transformation. We will make a change of variables from (x, y) to (p, ψ) as defined above. The elements of the Jacobian matrix of the transformation and its inverse are computed below. They relate the differential in X, Y to the differential in P, Ψ

$$\begin{pmatrix} dX \\ dY \end{pmatrix} = \begin{pmatrix} X_P & X_\Psi \\ Y_P & Y_\Psi \end{pmatrix} \begin{pmatrix} dP \\ d\Psi \end{pmatrix} = \mathbf{J}^{-1} \begin{pmatrix} dP \\ d\Psi \end{pmatrix}$$
$$\begin{pmatrix} dP \\ d\Psi \end{pmatrix} = \begin{pmatrix} P_x & P_y \\ \Psi_x & \Psi_y \end{pmatrix} \begin{pmatrix} dX \\ dY \end{pmatrix} = \mathbf{J} \begin{pmatrix} dP \\ d\Psi \end{pmatrix}.$$

Solving the second equation for dX, dY we get

$$\begin{pmatrix} dX \\ dY \end{pmatrix} = \frac{1}{P_x\Psi_y - P_y\Psi_x} \begin{pmatrix} \Psi_y & -P_y \\ -\Psi_x & P_x \end{pmatrix} \begin{pmatrix} dP \\ d\Psi \end{pmatrix}.$$

Comparing the two equations for dX, dY we find the elements of the Jacobian matrix and its inverse

$$\begin{array}{cccccccccccc} P_x & = & \frac{u}{\lambda K} & P_y & = & \frac{v}{\lambda K} & & X_P & = & \frac{u}{\|\mathbf{v}\|^2}\lambda K & X_\Psi & = & \frac{-v}{\|\mathbf{v}\|^2} \\ \Psi_x & = & -v & \Psi_y & = & u & & Y_P & = & \frac{v}{\|\mathbf{v}\|^2}\lambda K & Y_\Psi & = & \frac{u}{\|\mathbf{v}\|^2} \\ & & J = \frac{\|\mathbf{v}\|^2}{\lambda K} & & & & & & & J^{-1} = \|\mathbf{v}\|^{-2}\lambda K. & & & \end{array} \tag{3.2.1}$$

This is a time-dependent coordinate transformation so the saturation equation becomes

$$S_t - X_t S_x - Y_t S_y + \mathbf{v} \cdot \nabla f(S) = 0.$$

The second and third terms are due to the motion of the coordinates $p(t), \psi(t)$. Using the chain rule on the saturation we find

$$\begin{pmatrix} S_P \\ S_\Psi \end{pmatrix} = \begin{pmatrix} X_P & Y_P \\ X_\Psi & Y_\Psi \end{pmatrix} \begin{pmatrix} S_x \\ S_y \end{pmatrix}.$$

Solving this equation for S_x, S_y we can write the saturation equation in terms of the pressure and streamfunction variables

$$\begin{aligned} S_t - \frac{X_t}{J^{-1}}(Y_\Psi S_P - Y_P S_\Psi) - \frac{Y_t}{J^{-1}}(-X_\psi S_P + X_P S_\Psi) + \frac{\|\mathbf{v}\|^2}{\lambda K} f(S)_P &= 0 \\ S_t - \|\mathbf{v}\|^2 \frac{X_t Y_\Psi - Y_t X_\psi}{\lambda K} S_P - \|\mathbf{v}\|^2 \frac{Y_t X_P - X_t Y_P}{\lambda K} S_\Psi + \frac{\|\mathbf{v}\|^2}{\lambda K} f(S)_P &= 0. \end{aligned}$$

Note that in this formulation the flux term is one-dimensional. The advantages of the pressure-streamline frame in terms of simplicity and adaptivity are based on this fact. We can substitute the velocity in the last equation. Then the full equations in the pressure-streamline frame are

$$\begin{aligned} \nabla \lambda(S) K \nabla P &= 0 \\ \mathbf{v} &= \lambda(S) K \nabla P \\ \nabla^2 \Psi &= -\nabla^\perp \cdot \mathbf{v} \\ S_t - \tfrac{X_t u + Y_t v}{\lambda K} S_P - (Y_t u - X_t v) S_\Psi + \tfrac{\|\mathbf{v}\|^2}{\lambda K} f(S)_P &= 0. \end{aligned} \tag{3.2.2}$$

3.2.2. Entropy Solutions. In this section we demonstrate that the entropy solutions of the two-phase flow equations (2.1.16), (3.2.2) coincide. This is in general not obvious because the equation in the pressure-streamline frame is in nonconservative form. In our particular case the analysis follows easily from the fact that there exists a frame, namely the Cartesian frame, in which the equation is in conservation form. The first obstacle is that there is no proof of existence and uniqueness of solutions to the coupled two-phase flow equations. There is only a proof of existence of a solution when capillary forces, a diffusion term, are included in the equations [**1**]. The proof is made more complicated by the fact that the diffusion term is degenerate. When

$\lambda(S) = const$ the pressure equation is decoupled from the saturation and then we have a unique solution.

For the following we will assume one-phase flow. We will interpret the saturation equation in the sense of distributions, so we first derive that formulation for completeness. In the Cartesian frame the weak form of the saturation equation is

$$\int_{\mathbb{R}^2\times\mathbb{R}^+} (S\phi_t + f(S)\mathbf{v}\cdot\nabla\phi)dxdydt+ \\ + \int_{\mathbb{R}^2} S(x,y,t=0)\phi(x,y,t=0)dxdy = 0, \forall\phi(x,y,t)\in\mathcal{J}. \tag{3.2.3}$$

We change coordinates to arrive at the weak formulation in the pressure-streamline frame

$$\int_{\mathbb{R}^2\times\mathbb{R}^+} \left(S\phi_{t,\text{fix x}}+\right) J^{-1}dpd\psi dt+ \\ + \int_{\mathbb{R}^2} S(p,\psi,t=0)\phi(p,\psi,t=0)J^{-1}dpd\psi = 0, \forall\phi(p,\psi,t)\in C^\infty. \tag{3.2.4}$$

The Jacobian J^{-1} of the transformation was defined in (3.2.1). We have denoted with $\phi_{t,\text{fix x}}$ the derivative with respect to time, keeping x, y fixed. It is well known that when the flux function is nonlinear a hyperbolic equation can have many weak solutions. Motivated by the physical principle that entropy in nature cannot decrease when time increases we define admissible solutions to be the weak solutions that satisfy that property. To determine the correct entropy equation for our equations we start from the weak form of the entropy equation in the Cartesian frame. Let $\eta(\mu)$, $q(\mu)$ be an entropy-entropy flux pair. The following entropy condition must be satisfied for every convex entropy for the solution to be unique [**13, 34**]

$$\int_{\mathbb{R}^2\times\mathbb{R}^+} (\eta(S)\phi_t + q(S)\mathbf{v}\cdot\nabla\phi)\,dxdydt+ \\ + \int_{\mathbb{R}^2} \eta(S(x,y,t=0))\phi(x,y,t=0)dxdy \le 0, \forall \text{ nonnegative } \phi\in\mathcal{J}. \tag{3.2.5}$$

Changing variables we arrive at the correct entropy condition in the pressure-streamline frame

$$\int_{\mathbb{R}^2\times\mathbb{R}^+}\left(\eta(S)\phi_{t,\text{fix x}}+q(S)\frac{|\mathbf{v}|^2}{\lambda K}\phi_p\right)J^{-1}dpd\psi dt+$$
$$+\int_{\mathbb{R}^2}\eta(S(p,\psi,t=0))\phi(p,\psi,t=0)J^{-1}dpd\psi\geq 0,\forall \text{ nonnegative } \phi(p,\psi,t)\in\mathcal{J}$$

or

(3.2.6)
$$\int_{\mathbb{R}^2\times\mathbb{R}^+}\left(J^{-1}\eta(S)\phi_{t,\text{fix x}}+\right)dpd\psi dt+$$
$$+\int_{\mathbb{R}^2}\eta(S(p,\psi,t=0))\phi(p,\psi,t=0)J^{-1}dpd\psi\geq 0,\forall \text{ nonnegative } \phi(p,\psi,t)\in\mathcal{J}.$$

The strong form of the entropy condition is

(3.2.7)
$$\left(J^{-1}\eta(S)\right)_t-\frac{X_tu+Y_tv}{\lambda K}\left(J^{-1}\eta(S)\right)_p-(Y_tu-X_tv)\left(J^{-1}\eta(S)\right)_\psi+q(S)_p\leq 0.$$

The first three terms are the Lagrangian derivative of $J^{-1}\eta(S)$ in the moving frame $p(x,y,t),\psi(x,y,t)$.

The choice of the space $\mathcal{J}$ is not important for one-phase flow but for two-phase flow we must select it to be H_1, as we will see in the proof of the following theorem.

THEOREM 3.2.1. *Assume that the system of two-phase flow equations in the Cartesian frame (2.1.16) together with the entropy condition (3.2.5) interpreted in the sense of distributions $\phi\in H_1$ has a unique solution. Furthermore assume that the transformation $x,y\to p,\psi$ is nonsingular in the sense that the elements of the Jacobian matrix and its inverse are bounded. Then that solution coincides with the solution of the two-phase flow equations in the pressure-streamline frame (3.2.2) together with the entropy condition (3.2.7).*

PROOF. To show that $\phi(x,y)\in H_1\Leftrightarrow\phi(p,\psi)\in H_1$ we change variables in the definition of the H_1 norm and use the fact that the elements of the Jacobian matrix and its inverse are bounded. The transformation $x,y\to p,\psi$ is a bijection from H_1 to H_1 so S satisfies the weak pressure-streamline saturation equation (3.2.4) and

the associated weak entropy inequality(3.2.6) for all $\phi(p,\psi,t) \in C^\infty$ if (and only if) S satisfies the weak Cartesian saturation equation(3.2.3) and its associated weak entropy inequality (3.2.5) for all $\phi(x,y) \in C^\infty$. Thus the two solutions coincide and the uniqueness of the solution in the pressure-streamline frame follows from the uniqueness of the solution in the Cartesian frame. □

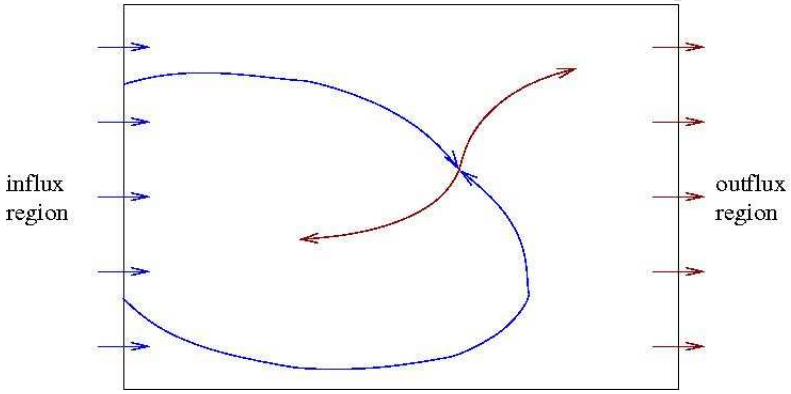
These results can be slightly modified to cover the more general case when there is no coordinate system where the equation is in conservation form and $S(\cdot,t), f(S)\mathbf{v}(\cdot,t)$ are in L_1 only. For the details we refer the interested reader to the work of LeFloch [**35**].

3.2.3. Invertibility. For the map to be invertible we need the Jacobian $\frac{\|\mathbf{v}\|^2}{\lambda K}$ to be positive. Since the permeability and λ are bounded from below we have to examine only the velocity. The requirement that the velocity be nonzero makes the method applicable for example only in flows where the streamlines are not closed because a closed streamline must contain a point with zero velocity . The maximum principle shows that there can be no sources or sinks inside the domain because that would result in an extremum of P and also a point where the divergence of the velocity field is not zero.

This doesn't exclude saddle points for the pressure. If we pick a convex domain with two sources of equal strength on the boundary so that they are opposite from each other, then along the line joining them there will be a point of zero velocity and a saddle point for the pressure. So it is the boundary condition that determines whether the velocity inside the domain is nonzero. If we assume that there are no sources or sinks on the $y = 0, 1$ boundaries then for flow from side-to-side there can be no saddle points. The two streamlines going into the saddle point would have to emanate from the $x = 0$ boundary, and the two streamlines that come out of the saddle point would have to end up at the sinks at $x = 1$. In two dimensions this cannot occur without streamlines crossing as is shown in figure 3.2.1. We believe that this argument can be made rigorous.

The analogous situation in three dimensions is when the domain of the problem is the unit cube with $x = 0$ as an influx and $x = 1$ as an outflux boundary and with

FIGURE 3.2.1. In 2D the flow has no saddle point



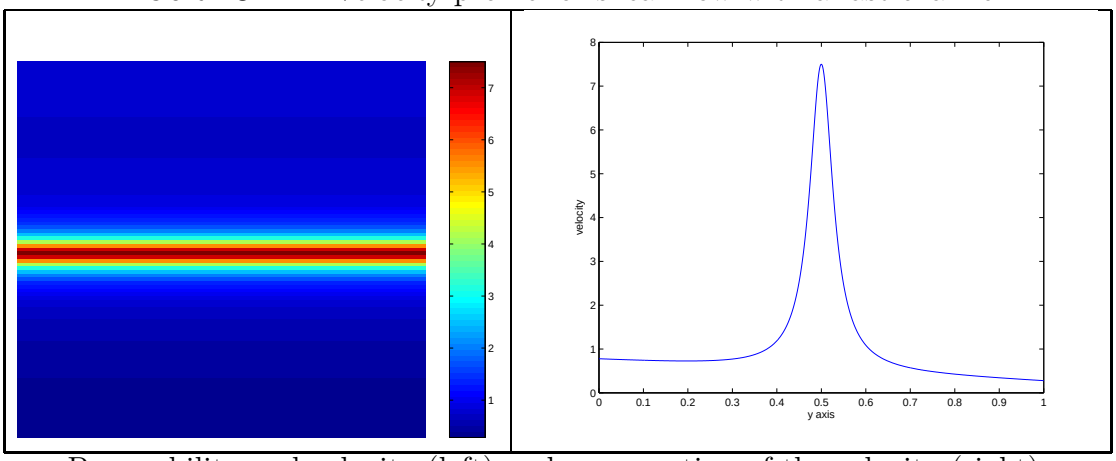
Both ougoing streamlines of a saddle point cannot end at the outflux boundary without crossing other streamlines.

the rest of the boundaries having no flux. In this case the boundary condition cannot guarantee that the Jacobian of the transformation does not become negative and we can only make progress by making an assumption that this does not happen.

3.2.4. Adaptivity. Rewriting the saturation equation using pressure and streamfunction as independent variables leads to a superior method in terms of adaptivity. Firstly, it accommodates the boundary conditions better than in the Cartesian frame. In a Cartesian frame we arbitrarily prescribe that there is no flux on the boundary of our domain. In realistic situations one would have to make a computation in a rectangular domain that includes all the regions where there is flow. In the pressure-streamline frame the domain of computation automatically covers the flow region. This leads naturally to a no-flow boundary condition at the boundary of the domain since that boundary is a streamline.

Adaptivity across the streamlines. Across the flow a mesh regularly spaced in the streamfunction variable ψ is automatically focused in regions with large gradients in the streamfunction variable. The magnitude of the gradient of ψ, the density of points in the pressure-streamline frame, is equal to the magnitude of the velocity, by (3.2.1), so the mesh is focused around the streamlines with larger velocity. More precisely, if we discretize the domain in cells uniformly spaced in p, ψ with spacing $\Delta p, \Delta\psi$ then every streamtube, defined as the region between $(\psi_0, \psi_0 + \Delta\psi)$, carries the same amount of fluid from the inlet boundary at $x = 0$ to the outlet at $x = 1$. The

FIGURE 3.2.2. Velocity profile for shear flow with a fast channel



Permeability and velocity (left) and cross section of the velocity (right).

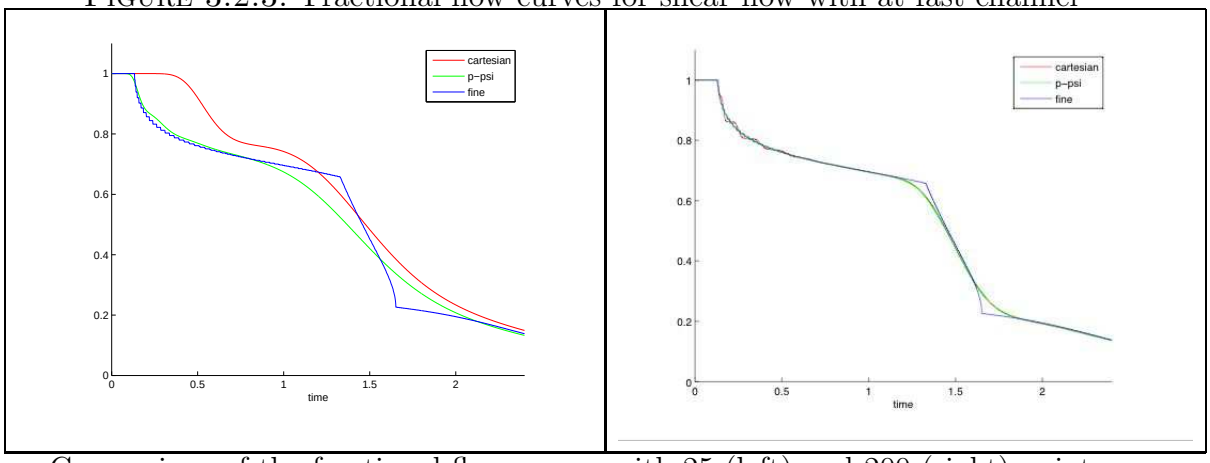
reason is that at the boundary of the streamtube there is no outward flux because it consists of streamlines and the velocity is tangent to them. Therefore the amount of fluid entering the streamtube at the inlet is the amount that exits at the outlet. That flux is equal to

$$\int_{\psi_0}^{\psi_0+\Delta\psi} \mathbf{v}(x = 0) \cdot \hat{\mathbf{x}} dy = \int_{\psi_0}^{\psi_0+\Delta\psi} u(x=0)\frac{dY}{d\psi}d\psi = \int_{\psi_0}^{\psi_0+\Delta\psi} d\psi = \Delta\psi.$$

In this derivation we used the fact that at $x = 0$, Ψ is only a function of Y. This expression quantifies the adaptivity principle with which we select the mesh along ψ.

We will demonstrate the advantage of selecting ψ as the variable across the flow in a simplified case of one-phase flow with linear flux. A permeability function that produces a flow with a fast channel is $K(x, y) = K(y) = 0.75 - 0.5y + \frac{7}{1+1000(y-0.5)^2}$. It is shown in figure 3.2.2. The streamlines in this case have a particularly simple structure; they are straight lines and the velocity is constant on each streamline. Since the permeability does not depend on x, the analytical solution for the pressure is $P = x$ and the velocity is $\mathbf{v} = (u, 0) = (K, 0)$. The saturation equation for the Cartesian and pressure-streamline frames is $S_t + uf(S)_x = 0$, where $u = u(y)$ or $u = u(\psi)$. The difference is that in the Cartesian frame the one-dimensional problems are regularly spaced in x and in the pressure-streamline frame they are regularly spaced in ψ. We see at the left part of figure 3.2.3, that the method in

FIGURE 3.2.3. Fractional flow curves for shear flow with at fast channel



Comparison of the fractional flow curves with 25 (left) and 200 (right) points.

the Cartesian frame does not resolve the fast flow channel well. The fractional flow rate has only one steep drop, which implies that if we select the streamlines in the Cartesian frame only one streamline lies on the fast flow channel. Hence the fractional flow rate is far from the exact solution. If we select the streamlines to be regularly spaced in ψ we obtain more accurate results, namely the fast channel is resolved better. The fast channel affects the fractional flow curve during early times $t < 1$ and is very well captured by the computation in ψ. The breakthrough time is more accurately predicted. At later times the fractional flow curve is less acurate because the slow regions are under-resolved by the computation in ψ.

Finally, the pressure-streamline coordinate frame eliminates cross-wind diffusion. Cross-wind diffusion is purely a numerical artifact associated with a non-adaptive choice for the coordinates in which we solve the equations. Consider the case of convection of the saturation in a Cartesian frame. The coordinate system is not aligned with the direction of the flow, and neither is the upwind direction. If we naively apply a time-splitting method, we are convecting the saturation first along $\mathbf{v} \cdot \hat{\mathbf{x}}$ and then along $\mathbf{v} \cdot \hat{\mathbf{y}}$ by upwinding in those two directions whereas the true upwind direction is along $\mathbf{v}$. The result is that the saturation is transported along the shock front in the cross-wind direction, besides across the shock front. Aside from the introduction of diffusion into the numerical solution, it is difficult to design second-order schemes with limiters that are stable and effective. In contrast, in a pressure-streamline frame, we have one-dimensional convection along the streamlines,

which means that there is no cross-wind diffusion, upwinding is accurate, and higher-order schemes are stable and easy to implement. For completeness we mention some efforts to design multidimensional upwind schemes. A first-order scheme that uses the correct upwind direction is the Corner Transport Upwind scheme, which was extended a second-order scheme by Colella [**12**] and van Leer [**51**]. A more recent effort can be found in [**18**].

Adaptivity along the streamlines. The pressure coordinate is adaptive in the sense that it places the points of computation near large velocities. To minimize the points of computation that are necessary for an accurate computation we use a moving mesh along the streamlines. The mesh will be concentrated near the regions where the numerical error is large, and it will be sparse in regions where the numerical error is small. For the Buckley Leverett equation we must resolve the shock well so that numerical diffusion will be small in the shock region. We make a transformation from a fixed coordinate system to a moving coordinate system $p(\xi, t)$ along the streamlines. The saturation equation becomes

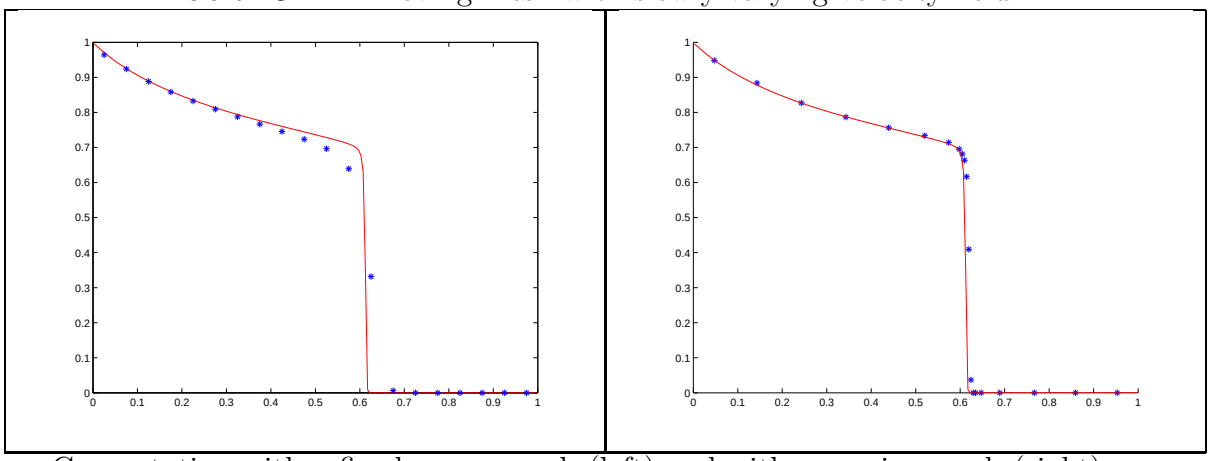
$$S_t - P_t S_p + v_0 f(S)_p = 0. \tag{3.2.8}$$

The first two terms are the Lagrangian time derivative, which is the time derivative of the saturation in the moving frame. In general the motion of the mesh is determined by a second partial differential equation, called the moving mesh partial differential equation (MMPDE). Our approach is based on the equidistribution principle first introduced by de Boor [**7**]. De Boor selected the variable $P(x, t)$ so that in each interval $\Delta\xi$ the arc length of the graph of the saturation function is the same, that is $\sqrt{\Delta S^2 + \Delta P^2} = C\Delta\xi$. The motion of the physical mesh is given by

$$\begin{aligned} \sqrt{S_p^2 + 1}\frac{\partial P}{\partial \xi} &= C \\ \frac{\partial}{\partial \xi}\sqrt{S_p^2 + 1}\frac{\partial P}{\partial \xi} &= 0. \end{aligned}$$

The quantity $w = \sqrt{S_p^2 + 1}$ is called the monitor function. Near the shock the arc length of the graph of the saturation increases rapidly, and the moving grid is concentrated. In general, we must select the monitor function to be a measure of the

FIGURE 3.2.4. Moving mesh with slowly varying velocity field



Computation with a fixed coarse mesh (left) and with a moving mesh (right).

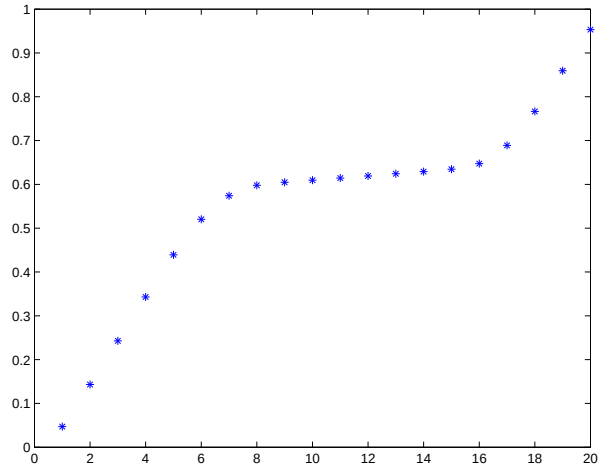
numerical error, so that the numerical error is equally distributed over the moving mesh points. We will use the second derivative in the monitor function and our MMPDE will be

$$\frac{\partial}{\partial \xi}\sqrt{S_{pp}^2 + E^2}\frac{\partial P}{\partial \xi} = 0 \tag{3.2.9}$$

where E is a constant. Equations (3.2.8), (3.2.9) form a system for S, P. A classification of different MMPDEs and some of their properties can be found for example in papers by Huang and Russell [**32**].

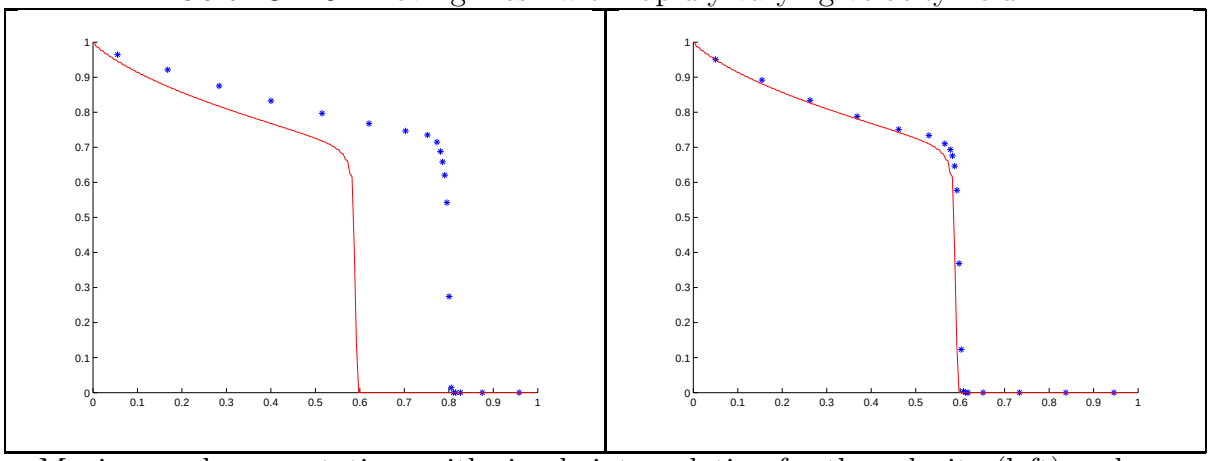
To demonstrate numerically the adaptivity of a moving mesh we solve the Buckley Leverett equation with $m = 1$ with a moving mesh and without. The fine computation has 200 cells and the coarse computation 20 cells. The minimum mesh spacing for the moving mesh that is allowed was selected to be the mesh spacing of the fine computation, that is $h_{min} = 0.005$. If we allowed the mesh to cluster more we would obtain a sharper shock at the expense of a longer computation. The velocity field was $v(p) = 1 - 0.2cos(2\pi p)$ on $[0, 1]$. In figure 3.2.4 we show the saturation at $t = 0.5$. Besides the gain in complexity which is a factor of 10 we see a sharper shock profile because the mesh has been concentrated near the shock. In figure 3.2.5 we show a plot of the map at the final time demonstrating that the mesh is concentrated near the shock.

FIGURE 3.2.5. The mesh transformation



The moving mesh points are concentrated near the shock.

FIGURE 3.2.6. Moving mesh with rapidly varying velocity field



Moving mesh computations with simple interpolation for the velocity (left) and interpolation of the geometric average (right).

We note that the moving mesh equations are valid when the velocity field is smooth on the coarse scale of the moving mesh equations. If the velocity field varies on the fine scale then when we interpolate the velocity onto the moving mesh points we make a large error. For an accurate computation in this case we have to replace the velocity with its geometric average in the moving mesh equations and interpolate the geometric average of the velocity onto the moving mesh points instead. Results for a velocity field that is equal to 1 and 4 alternatively on the fine grid on $[0, 1]$ at time $t = 0.3$ are in figure 3.2.6.

We provide insight into why this occurs and a proof for convergence in part two of this work. We will show that the effective velocity of transport over a region is given by the geometric average of the velocity in that region.

Adaptivity in time. Decoupling the full problem into one-dimensional problems facilitates making the algorithm adaptive in time. It is easy to take a different time step along every channel. To make the method even more adaptive in time we can use locally adaptive time steps along the channels as well, see for example [**48**].

3.2.5. Extension to Three Dimensions. We will consider flows in the unit square three dimensions. We will assume that $P(x = 0) = 0, P(x = 1) = 1$ and a Neumann condition for the rest of the boundary. The construction of the map in three dimensions is slightly more complicated than in two dimensions. The reason is that the normal vector to the velocity is not unique. The primary reason why we selected p, ψ is because the saturation equation reduces to a one-dimensional equation with advantages in adaptivity and simplicity. We transform the flux term of the saturation equation in an arbitrary coordinate system (p, ψ_1, ψ_2)

$$\mathbf{v} \cdot \nabla f(S) = \mathbf{v} \cdot \nabla P f(S)_p + \mathbf{v} \cdot \nabla \Psi_1 f(S)_{\psi_1} + \mathbf{v} \cdot \nabla \Psi_2 f(S)_{\psi_2}.$$

To have the same outcome in three dimensions we need a coordinate system with one direction along the flow and all other directions perpendicular to the flow. This requirement results in a unique two-dimensional map but not in unique higher-dimensional maps.

Thus we can construct the map $x, y, z \to p, \psi_1, \psi_2$, by selecting ψ_1, ψ_2 to be orthogonal to p. An excellent discussion of the streamfunction in three dimensions can be found in [**5**]. To focus the computational grid close to the fast streamlines we impose a constraint on how ψ_1, ψ_2 are selected. Following the same line of thought as in two dimensions, we define a streamtube to consist of the points with $(\psi_1, \psi_2) \in (\psi_{10} + \Delta\psi_1, \psi_{20} + \Delta\psi_2)$. Since the velocity is tangent to the boundary of the streamtubes inside the domain, the flux at the inlet of the streamtube equals the flux at the outlet. At $x = 0$, the inlet, we consider $\psi_1, \psi_2 \to y, z$ as a two-dimensional transformation, and we impose that its Jacobian be the magnitude of the velocity.

This means that the area in the ψ_1, ψ_2 coordinates is equal to the flux at any point into the domain, that is

$$d\psi_1 d\psi_2 = |\mathbf{v}| dy dz.$$

Using this relation we can show that each streatube in the p, ψ_1, ψ_2 domain carries the same amount of fluid

$$\begin{aligned}\int_{\psi_{10}}^{\psi_{10}+\Delta\psi_1} \int_{\psi_{20}}^{\psi_{20}+\Delta\psi_2} \mathbf{v}(x=0)\cdot\hat{\mathbf{x}} dy dz &= \int_{\psi_{10}}^{\psi_{10}+\Delta\psi_1} \int_{\psi_{20}}^{\psi_{20}+\Delta\psi_2} \mathbf{v}\cdot\hat{\mathbf{x}} dy dz \\ = \int_{\psi_{10}}^{\psi_{10}+\Delta\psi_1} \int_{\psi_{20}}^{\psi_{20}+\Delta\psi_2} u\frac{d\psi_1 d\psi_2}{|\mathbf{v}|} &= \int_{\psi_{10}}^{\psi_{10}+\Delta\psi_1} \int_{\psi_{20}}^{\psi_{20}+\Delta\psi_2} d\psi_1 d\psi_2 \\ &= \Delta\psi_1 \Delta\psi_2.\end{aligned}$$

After we select the parametrization of the $p = 0$ or $x = 0$ surface in terms of ψ_1, ψ_2, we convect the values of the parameters with the velocity field to the rest of the level sets of the pressure.

3.3. Numerical Implementation

3.3.1. Computing Ψ,v_0. After solving for the pressure we need to extract Ψ and then v_0, which will be used in the saturation equation. Instead of solving the Poisson equation (2.1.16) to compute Ψ we will use the first-order equation (3.2.1). This is faster and more accurate. Since the boundary $y = 0$ is a streamline we set $\Psi(x, y = 0) = 0$ and integrate along y

$$\Psi(x, y) = \int_0^y u(x, s) ds.$$

On the continuous level the fact that the velocity field is conservative guarantees that the line integral $\int_{\partial\Omega} \nabla\Psi \cdot \mathbf{t} dt$, where $\mathbf{t}$ is the tangent vector to $\partial\Omega$, vanishes over any subdomain of the unit disk. This makes Ψ single valued on any point of the domain. On the discrete level we assume that the velocity is piecewise constant on the edges of the dual control volumes. The fact that the discrete field satisfies the discrete conservation condition means that we can integrate along any path that consists of boundaries of dual control volumes to compute Ψ. We will simply integrate

along $x = ih$

$$\Psi_{i,j} = \sum_{s=1}^{j} u_{i,s} h.$$

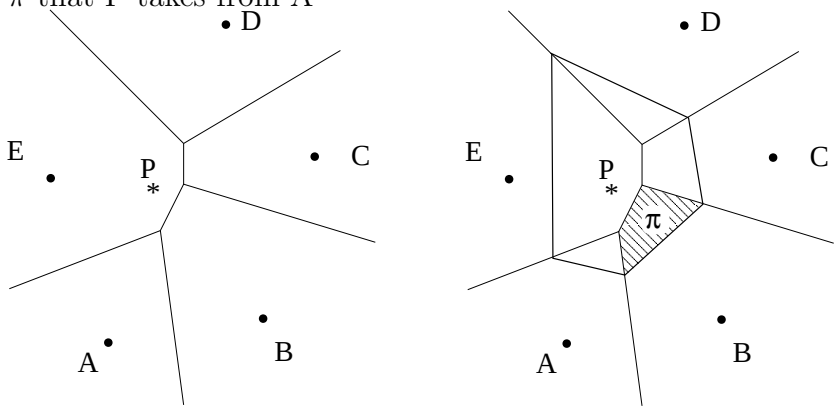
More importantly, discrete conservation implies that $\Psi_{i,j}$ computed with the formula above will have the same value at $y = 1$, and our transformation maps the unit square onto a rectangular domain. Once we compute Ψ at the edges of the primal grid, we linearly interpolate it to the edges of the dual grid.

To compute v_0 we interpolate the quantities $\frac{u^2}{\lambda K}, \frac{v^2}{\lambda K}$ from the edges of the dual grid cell to its center. We could have avoided having to interpolate the components of the velocity to the same point to obtain its magnitude by using triangular cells instead of quadrilateral cells. Then we would know ∇P in the center of the dual triangles and we would only have to interpolate the permeability. This would eliminate a source of error, but would complicate the implementation without illustrating the principles any better. Next we have to interpolate the velocity and saturation to the pressure-streamline frame.

3.3.2. Natural Neighbor Coordinates. To obtain quantities on a regular grid in p, ψ from a regular grid in x, y involves interpolation over an irregular grid. We are given a set of points $p_{i,j}, \psi_{i,j}$ and function values $S_{i,j}$, and we want to interpolate them linearly to the points $p = ih$ and $\psi = jh$. Let p^*, ψ^* be a point on which we want to interpolate the values S_{ij}. If the grid were regular we would use the vertices of the cell in which p^*, ψ^* lies to reconstruct S^*. When the grid is irregular there is no unique way to decide which of the points in the neighborhood of p^*, ψ^* should be used to construct S^*, that is, who the neighbors of p^*, ψ^* are. If we choose the wrong neighbors the interpolation error can be large.

We briefly introduce the concepts of Voronoi diagrams and natural neighbor coordinates. A Voronoi diagram is a partition of the domain Ω that associates a cell $\Omega_{i,j}$ to every point $p_{i,j}, \psi_{i,j}$ of an irregular grid, according to the following rule. A point belongs to $\Omega_{i,j}$ if it is closer to $p_{i,j}, \psi_{i,j}$ than any other point of the grid. We consider the Voronoi diagram of $p_{i,j}, \psi_{i,j}$ and p^*, ψ^*. The neighbors of $p^*, \psi^*$are the points whose Voronoi cells have a common edge with the cell of p^*, ψ^*. Note that the number of neighbors is not fixed for all neighbors, but depends on the local shape

FIGURE 3.3.1. The interpolation weight of A is proportional to the area π that P takes from A



of the grid. The interpolant at p^*, ψ^* is the weighted average of the neighbors. The weight of neighbor, called the natural neighbor coordinate $p_{i,j}, \psi_{i,j}$, is the ratio of the area that p^*, ψ^* takes from the cell of $p_{i,j}, \psi_{i,j}$ when p^*, ψ^* is added to the Voronoi diagram of $p_{i,j}, \psi_{i,j}$,to the area of the cell of p^*, ψ^*. This is illustrated in figure 3.3.1.

This idea was first proposed by Sibson [**46**]. It can be shown that the interpolant defined by this procedure is linear. We observed that when the computational domain is $(0,1) \times (0,c)$ natural neighbor coordinates lead to a large number of neighbors and a lot of numerical diffusion. We can change the aspect ratio of the computational domain, which will lead to less singular triangles and less diffusion. In practice we observe that the most accurate interpolation is the one where the neighbors of a point P^*, Ψ^* are the vertices of the quadrilateral that encloses P^*, Ψ^*. To find their interpolant we use natural neighbor interpolation described above.

With this method we can interpolate the saturation and the velocity onto a regular grid in p, ψ. It will guarantee that the variation of the saturation does not increase when we move to the pressure-streamline frame since linear interpolation is averaging; therefore the numerical error is some extra diffusion in the saturation. This diffusion will guarantee that the whole numerical scheme for the saturation remains Total Variation Diminishing.

For the velocity it is equally reasonable to use linear or quadratic interpolation. The velocity interpoplation is most crucial and seems to be the source of most numerical error. We note here that whereas the pressure is smooth the velocity has

$O(1)$ oscillations over length $O(\epsilon)$. We implemented Sibson's quadratic interpolant but didn't notice a significant difference.

3.3.3. Finite Volume Scheme for the Saturation in p, ψ. As in the numerical method for the Cartesian frame, we will use an IMPES method for the time discretization of the equations. By using the IMPES scheme we don't have to deal with the second and third terms of (3.2.2). These terms depend on the change of the streamlines and by using a first-order discretization of the velocity in time, which depends on the pressure, we are assuming that the streamlines remain constant during a pressure time step. The full saturation equation (3.2.2) in an IMPES framework without these terms is

$$S_t + v_0 f(S)_p = 0. \tag{3.3.1}$$

To discretize this equation correctly we consider its physical interpretation. $S(p, \psi)$ is not a conserved quantity, as is obvious by (3.3.1). We can derive a conserved quantity from the conserved quantity in the Cartesian frame using (3.2.1)

$$\int S dx dy = \int S J^{-1} dp d\psi = \int \frac{S}{v_0} dp d\psi.$$

We can arrive at the same conserved quantity by changing coordinates in the weak formulation of the conservation law. We arrive at a second interpretation of $v_0(x, y)$ as the capacity of the streamline at x, y. We have decomposed the two-dimensional convection into one-dimensional convection along streamlines, each with a given capacity at every point. We can also think of streamtubes and their width. We divide (3.3.1) by v_0 and integrate over a small volume to arrive at the integral conservation form. If S_i^t denotes the average of the saturation in the ith volume at time t and $\frac{1}{v_{0i}}$ the capacity of that cell, we obtain

$$\frac{S_i^{t+\Delta t}}{v_{0i}} = \frac{S_i^t}{v_{0i}} - \frac{\Delta t}{\Delta p} \left(\mathfrak{F}(S_i^t, S_{i+1}^t) - \mathfrak{F}(S_{i-1}^t, S_i^t) \right).$$

An integer subscript i denotes a quantity at the center of a cell or an averaged quantity over a cell, and a subscript $i - 1/2$ denotes a quantity defined at the edge between the cells with indices i and $i - 1$.

In general, the flux $\mathfrak{F}$ describes the effect of a right- and a left-going wave on the average saturation S_i. We will use a Reconstruct-Solve-Average (RSA) approach and will average the saturation profile over cells at the beginning of the time step, advance the Riemann problems for a time Δt, and then average the saturation again. For the Buckley Leverett equation (3.3.1) there is only a right-going wave of strength $W_{i-1/2}$ and speed $\sigma_{i-1/2}$ given by

$$\begin{aligned} W_{i-1/2} &= S_i - S_{i-1} \\ \sigma_{i-1/2} &= \begin{cases} \frac{f(S_i)-f(S_{i-1})}{S_i-S_{i-1}} & S_{i-1} \neq S_i \\ f'(S_i) & S_{i-1} = S_i \end{cases} . \end{aligned}$$

Godunov's method will have

$$\mathfrak{F}(S_{i-1}, S_i) = \sigma_{i-1/2} W_{i-1/2} = f(S_i) - f(S_{i-1}).$$

We will use a second-order method based on the Lax Wendroff scheme. It can be derived by Taylor expanding $S^{t+\Delta t}$ in time and substituting (3.3.1) for the time derivatives. It can be considered as Godunov's scheme with a flux correction F. The full scheme is given by

$$\begin{aligned} \tilde{W}_{i-1/2} &= \phi(\theta_{i-1/2}) W_{i-1/2} \\ F_{i-1/2} &= \tfrac{1}{2}\sigma_{i-1/2}(1 - \tfrac{1}{v_{0i-1/2}}\tfrac{\Delta t}{\Delta p}\sigma_{i-1/2})\tilde{W}_{i-1/2} \\ S^{t+\Delta t} &= S_i^t \quad -v_{0i}\tfrac{\Delta t}{\Delta p}(f(S_i) - f(S_{i-1})) - v_{0i}\tfrac{\Delta t}{\Delta p}(F_{i+1/2} - F_{i-1/2}), \end{aligned} \tag{3.3.2}$$

with $\theta_{i-1/2} = \frac{S_{i+1}-S_i}{S_i-S_{i-1}}$. To ensure that the scheme remains Total Variation Diminishing (TVD) we employ a minmod limiter, which is described above by ϕ. In the flux correction $F_{i-1/2}$, $\frac{1}{v_{0i-1/2}}$ can be selected as the cell-centered capacity $\frac{1}{v_{0i}}$ or $\frac{1}{v_{0i-1}}$, or as their average, and the numerical method will still be second-order accurate. We will select the latter motivated by the fact that the second time derivative of the saturation derived from (3.3.1)

$$S_{tt} = v_0 \left(v_0 \left(f'(S)^2 \right) S_p \right)_p$$

TABLE 1. 2nd order convergence of fine p, ψ scheme

	25	50	100	200	400
L_1 relative error	5.66×10^{-4}	1.64×10^{-4}	4.43×10^{-5}	1.15×10^{-5}	2.89×10^{-6}
ratio of L_1 errors		3.45	3.7	3.85	3.98
L_∞relative error	0.0015	4.88×10^{-4}	1.76×10^{-4}	4.78×10^{-5}	1.27×10^{-5}
ratio of L_∞ errors		3.1	2.77	3.68	3.76

TABLE 2. 1st order convergence of fine p, ψ scheme near extrema

	25	50	100	200	400	800
L_1 relative error	0.0022	8.92×10^{-4}	3.30×10^{-4}	1.22×10^{-4}	4.64×10^{-5}	1.80×10^{-5}
ratio of L_1 errors		2.47	2.70	2.70	2.62	2.57
L_∞ relative error	0.0074	0.0049	0.0033	0.0020	0.0011	6.06×10^{-4}
ratio of L_∞ errors		1.5	1.48	1.65	1.82	1.82

contains v_0 evaluated between S_i and S_{i-1} when discretised with centered differences, as in the Lax Wendroff scheme. The CFL condition is

$$\Delta t \leq \frac{\Delta p}{f'(S_i) v_{0i}}.$$

We will demonstrate the accuracy of our method in the case of nonlinear flux with $m = 5$. For the computation of the error we used a simulation with 4000 cells. For $S(t = 0) = 1 - 0.9p^2$, $v_0 = 1 + 0.5 \sin(5\pi p)$, at $t = 0.3$ we obtained the errors of table 1. We can see that as we refine the grid the order of convergence, the base 2 logarithm of the ratio of the errors, tends to 2. To test that the limiter does not introduce too much error we select an initial condition with an extremum, $S(t = 0) = 0.55 - 0.05 \frac{\|p - 0.5\|}{0.5}$, $v_0 = 1 + 0.5 \sin(5\pi p)$. This initial condition varies slowly so it can be resolved by a small number of points. It has a large second derivative at the extremum, and therefore a first-order method will have a large error near the extremum, which allows us to observe the error convergence clearly. At $t = 0.3$ we obtain the errors in table 2.

At the extremum of the saturation, the scheme is reduced to first-order because of the limiter so the method is only first-order in L_1. If we continue refining the grid we will see the ratio of the L_1 norms converging to 4. We note here that the observed order of the scheme is close to the theoretical predictions only for very smooth, that is, very well resolved, saturation profiles and velocities. We report the errors for the

TABLE 3. Order of accuracy of fine p, ψ scheme in the presence of shocks

	25	50	100	200	400
L_1 relative error	0.0400	0.0213	0.0114	0.0060	0.0030
ratio of L_1 errors		1.88	1.87	1.9	2.0
L_∞ relative error	0.257	0.265	0.258	0.261	0.260
ratio of L_∞ errors		N/A	N/A	N/A	N/A

same velocity field, $v_0 = 1 + 0.5\sin(5\pi p)$, when the initial condition is a Heaviside function, at $t = 0.5$, in table 3. In the presence of a shock the method is only first-order with respect to the L_1 norm, and the L_∞ error does not converge. The error is the same as the interpolation error of a discontinuous function on a uniform grid.

3.3.4. Moving Mesh. There are two approaches to solving moving mesh equations (3.2.8), (3.2.9). The first is to solve them in the physical domain p and interpolate the solution to the new mesh at every time step, and the other is to write in terms of computational variables and solve them in the computational domain ξ. The first has the advantage of being in a conservation form; however the interpolation can lead to numerical errors. For the second, no interpolation is required, but the equation that must be solved is more complicated and not in conservation form and that can be a source of numerical error.

For conservation laws there is a strong reason to prefer the physical domain. The numerical method for the saturation equation can be interpreted as solving a Riemann problem for each cell and then interpolating the solution to a fixed grid. Since there is already an interpolation involved in solving the physical equation, there is a lot of knowledge about the interpolation errors and methods (ENO, limiters) to reduce it effectively. Moreover an extra interpolation fits naturally in the framework of a method that already contains an interpolation.

In practice, instead of solving the moving mesh equation (3.2.9) we simply compute the values of the monitor function w_i associated with saturation values S_i^n and grid point locations P_i^n at the previous time step and then interpolate the inverse of the monitor function $P_i^n(w_i)$ onto a uniformly spaced grid in w. The resulting values P_i are the new mesh locations at the next time step. To compute the monitor function

we first compute the second derivative of the saturation using finite differences. To avoid very singular monitor functions we pass the result through a first-order Fourier filter, a few Jacobi iterations of the heat equation. The effect is to damp the high frequency modes of the monitor function. In theory once we have found the optimal number of filter passes, when we double the number of moving mesh points we should double the number of filter passes so that the smoothing of the monitor function is the same. In practice the increased accuracy does not compensate enough for the increased computational cost because the monitor function is not accurate enough, so in our algorithm we kept the number of filter passes fixed. In regions where the second derivative vanishes the mesh can be very sparse and the error can be large. To remedy this, we don't allow the monitor to be smaller than E, which we select to be

$$E = \frac{h_{min}}{h_{MM}} \max S_{pp}. \tag{3.3.3}$$

Since E is the minimum of the monitor function, this choice for E imposes that $\frac{h_{min}}{h_{MM}}$ be the desired ratio between the minimum and maximum spacing of the moving mesh grid. We will impose that the minimum spacing be h_{min} in the next step of the method. In this step we select the maximum spacing of the moving mesh points to be $h_{MM} = \frac{1}{\text{number of moving mesh points}}$. In regions where the mesh is concentrated the derivative P_ξ is small and the CFL condition

$$\Delta t \leq \frac{P_\xi}{v f'(S)} \Delta\xi$$

leads to a severe time step constraint. After we determine E by(3.3.3) we impose a cutoff to the monitor function so that the minimum mesh spacing is h_{min}. The moving mesh solution can be at best as accurate as a uniform computation with grid spacing h_{min}. When comparing an upscaled and a fine computation we will often take h_{min} to be the mesh size of the fine grid. We observed that the moving mesh points oscillate around their trajectories. This is because in the computation of the monitor function at time $t + \Delta t$ we use the saturation at time t and not at time $t + \Delta t$. To eliminate these oscillations we use the fact that the general mesh motion is to the right. When the grid point moves to the left we filter it by $x(t + \Delta t) \to 0.95x(t) + 0.05x(t + \Delta t)$.

When we interpolate the saturation from the grid points of the previous time step to the next time step it is important to obtain the initial condition for the saturation equation so that the total variation does not increase and the scheme remains conservative. Otherwise there is no guarantee that the numerical scheme will converge. We overcome these difficulties by interpolating through advancing the following equation

$$S_t^* - P_t S_p^* = 0.$$

This idea was first proposed by Li, Tang, Zhang [**37**]. The initial condition is $S^*(t_n) = S^n(x^n)$, and the resulting saturation is the interpolated saturation on the new mesh, $S^*(t_{n+1}) = S^n(x^{n+1})$, which is then used as an initial condition for the saturation equation. The full scheme can be viewed as an operator splitting scheme

$$S_t - P_t S_p + v_0 f(S)_p = 0 \longmapsto \left\{ \begin{array}{rcl} S_t + v_0 f(S)_p & = & 0 \\ S_t - P_t S_p & = & 0 \end{array} \right\}. \tag{3.3.4}$$

We note that the split scheme retains second-order accuracy, provided that we solve both parts of (3.3.4) with a method that is at least second-order accurate. That is because the differential operators that correspond to (3.3.4) commute. To see this we can write the second equation in the static coordinate frame, $S_t = 0$, from which it becomes clear that it has no physical effect on the solution. We discretise the top equation of the time-splitting scheme (3.3.4) on an irregular grid P_i, following the same line of thought as in the previous section. Define P_{i+1}, P_1 to be the boundaries of the ith cell, $\kappa_i = \frac{1}{2}(P_{i+1} - P_i)$ to be the area of the cell, and $\kappa_{i-1/2} = \frac{1}{2}(\kappa_{i+1} - \kappa_i)$. The Lax Wendroff scheme (3.3.2) becomes

$$\begin{array}{rcl} \tilde{W}_{i-1/2} & = & \phi(\theta_{i-1/2}) W_{i-1/2} \\ F_{i-1/2} & = & \frac{1}{2}\sigma_{i-1/2}(\frac{\kappa_{i-1}}{\kappa_{i-1/2}} - \frac{1}{v_{0i-1/2}}\frac{\Delta t}{\kappa_{i-1/2}\Delta\xi}\sigma_{i-1/2})\tilde{W}_{i-1/2} \\ S^{t+\Delta t} & = & S_i^t - v_{0i}\frac{\Delta t}{\kappa_i \Delta\xi}(f(S_i) - f(S_{i-1})) - v_{0i}\frac{\Delta t}{\kappa_i\Delta\xi}(F_{i+1/2} - F_{i-1/2}). \end{array}$$

For the correct upwinding of the interpolation equation of (3.3.4), we rewrite the convection term as a conservative convection term and a forcing

$$S_t - (P_t S)_p + P_{tp} S = 0. \tag{3.3.5}$$

The moving mesh velocity is linear in each cell so the forcing term can be integrated exactly, given the moving mesh velocity at the edges of the cells. The first-order wave is easily determined by (3.3.5). To find the correction we follow the derivation of the Lax Wendroff scheme and Taylor expand the saturation in time, and substitute the time derivatives using (3.3.5)

$$\begin{aligned} S(t^{n+1}) &= S(t^n) + \Delta t S_t(t^n) + \frac{1}{2}\Delta t^2 S_{tt}(t^n) \\ &= S(t^n) - \Delta t \left(P_t S(t^n)\right)_p + \Delta t P_{tp} S(t^n) + \frac{1}{2}\Delta t^2 \left(P_t^2 S_p(t^n)\right)_p + \frac{1}{2}\Delta t^2 P_{tp} S_t(t^n). \end{aligned}$$

The numerical scheme is given by

$$\begin{aligned} \tilde{W}_{i-1/2} &= \phi(\theta_{i-1/2}) W_{i-1/2} \\ F_{i-1/2} &= \tfrac{1}{2} P_{ti-1}(\tfrac{\kappa_{I-1}}{\kappa_{i-1/2}} - \tfrac{\Delta t}{\kappa_{i-1/2}\Delta\xi} P_{ti-1})\tilde{W}_{i-1/2} \\ S^{t+\Delta t} &= S_i^t - \tfrac{\Delta t}{\kappa_i \Delta\xi}(P^+_{ti-1} W^+_{i-1/2} + P^-_{ti} W_{i+1/2} - (P_{ti} - P_{ti-1}) S_i^t) - \\ &\quad - \tfrac{\Delta t}{\kappa_i \Delta\xi}(F_{i+1/2} - F_{i-1/2} + \tfrac{\Delta t}{\kappa_{i-1/2}\Delta\xi}(P_{ti}^2 - P_{ti-1}^2) W_{i-1/2}), \end{aligned}$$

where f^+ is defined as $\max\{f, 0\}$ and similarly for f^- and where $\theta_{i-1/2} = \frac{S_{i+1} - S_i}{S_i - S_{i-1}}$, $I = i - 1$ if $P_{ti} > 0$, and $\theta_{i-1/2} = \frac{S_{i-1} - S_{i-2}}{S_i - S_{i-1}}$, $I = i$ if $P_{ti} < 0$.

To demonstrate the order of accuracy of a moving mesh method we must design a numerical experiment that satisfies two contradictory conditions. The initial condition and the velocity must be well resolved by the grid so that the rate of error convergence has reached 2 and at the same time the saturation profile must be varying enough so that the mesh moves and our experiments are meaningful. We select the same conditions that we used to show second-order convergence for the scheme on the fixed grid, $S(t = 0) = 1 - 0.9p^2$, $v_0 = 1 + 0.5\sin(5\pi p)$, at $t = 0.3$. The mesh was not moved using the moving mesh equation, which would result in little mesh motion, but according to $P(t, p) = p + 0.1\frac{0.3 - t}{0.3}\sin(2\pi p)$. The errors at $t = 0.3$ are shown in table 4 and we observe second-order convergence.

To demonstrate the accuracy of the moving mesh we repeat the experiment with a shock, $v_0 = 1 + 0.5\sin(5\pi p)$, and a Heaviside function as initial condition. First we look at the case of linear flux. We filter the monitor function 4 times and choose $E = 0.05\frac{h_{min}}{h_{MM}}\max S_{pp}$. This means that the maximum spacing allowed is very large.

TABLE 4. Order of accuracy of p, ψ scheme with moving mesh

	25	50	100	200	400
L_1 relative error	0.0013	3.67×10^{-4}	9.38×10^{-4}	2.39×10^{-5}	6.02×10^{-6}
ratio of L_1 errors		3.54	3.9	3.92	3.97
L_∞relative error	0.0041	0.0012	3.07×10^{-4}	7.76×10^{-5}	1.95×10^{-5}
ratio of L_∞ errors		3.42	3.91	3.96	3.98

TABLE 5. Efficiency of moving mesh for linear flux

	uniform	cost		$h_{min} = \frac{h_{MM}}{10}$	cost	$h_{min} = \frac{h_{MM}}{20}$	cost	$h_{min} = \frac{h_{MM}}{40}$	cost
100	0.0486	2	25	0.0502	2	0.0532	5	0.0396	8
200	0.0324	6	50	0.0239	10	0.0202	20	0.0217	37
400	0.0207	19	100	0.119	39	0.0083	78	0.0058	148
800	0.0128	75	200	0.0064	145	0.0040	286	0.0024	578
1600	0.0077	310	400	0.0034	538	0.0018	1067	8.08×10^{-4}	2209
3200	0.0044	1226							

The first three columns are the number of grid points, the L_1 relative error, and the computational cost for the uniform grid. The rest of the columns are the same quantities for the moving mesh algorithm, for three different choices for h_{min}.

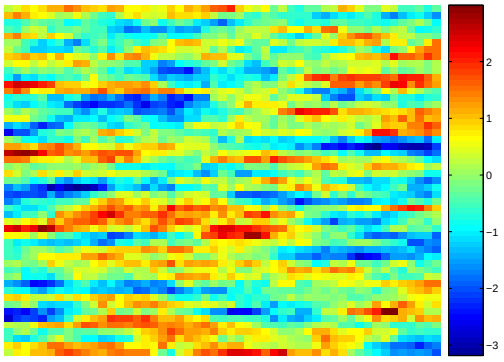
We can afford to do this because away from the shock the exact solution is a constant and can be resolved with very few points. The errors and times of computation at $t = 1.0$ are shown in table 5. The optimal moving mesh computation is the one with $h_{min} = \frac{h_{MM}}{40}$. If we extrapolate the entries for the uniform computation we see that a moving mesh computation with 200 points has the same error as a computation of 6400 uniform points, a factor of 32. The moving mesh computation is 8 times faster. In table 5 when the number of points is doubled the error is halved and the computational cost is quadrupled because the algorithm has twice as many points in space and twice as many time steps. To have the error using a moving mesh we can also halve h_{min}. This doubles the number of time steps but does not double the grid points. It simply uses them more efficiently so the cost is only doubled. This is the case for using a moving mesh.

We repeat the experiments for the Buckley Leverett flux with $m = 1$. We filter the monitor function 8 times and choose $E = 0.05 \frac{h_{min}}{h_{MM}} \max S_{pp}$. The results at $t = 0.7$ are shown in table 6. The results are not so impressive in this case. In most computations we observe that for the same L_1 error it takes a uniform grid with 4 times as many points as the moving mesh grid, but the computational cost is the same. The reason

TABLE 6. Efficiency of moving mesh for nonlinear flux

uniform mesh	L_1 relative error	cost	moving mesh	L_1 relative error	cost
100	0.0102	2	25	0.0110	4
200	0.0051	7	50	0.0041	16
400	0.0025	27	100	0.0022	59
800	0.0011	107	200	8.9×10^{-4}	216
1600	5×10^{-4}	428	400	3.45×10^{-4}	894
3200	2.5×10^{-4}	1712			

FIGURE 3.4.1. 2pt geostatistics permeability field



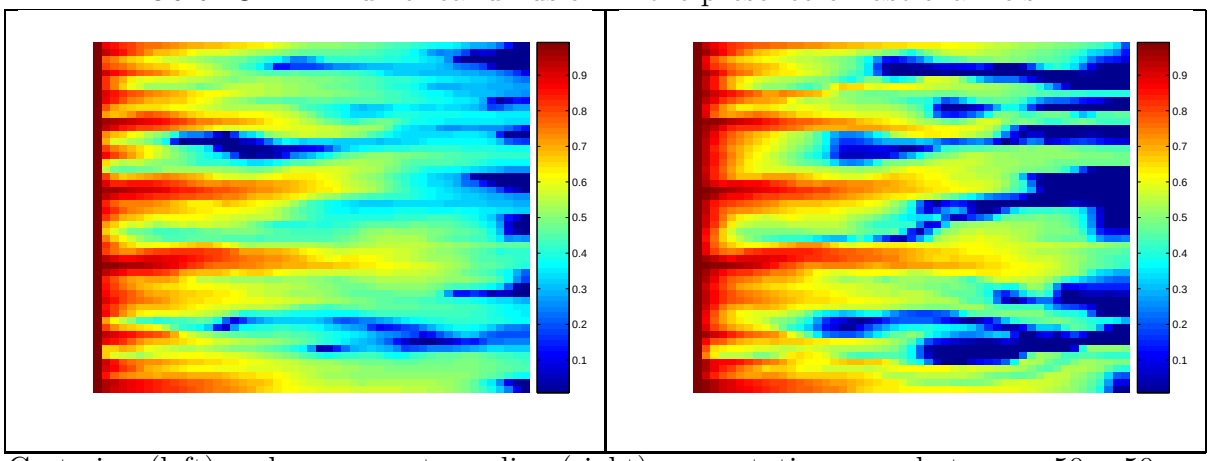
Plot of the permeability in a logarithmic scale.

is that now we need to resolve the region away from the shock to have an accurate shock speed, so the moving mesh can be focused less near the shock. If we compare the uniform computations in tables 5, 6 the accuracy in the nonlinear case increases. With a linear flux we have a contact discontinuity and all the characteristics in the region of the shock are parallel, whereas with the nonlinear flux we have a shock and the characteristics flow into the shock. This reduces the numerical diffusion in the nonlinear case and makes the uniform computation more accurate; hence less points are needed for a sharp shock.

3.4. Numerical Results

We want to demonstrate the superiority in terms of adaptivity of a method in the pressure-streamline frame using realistic permeability fields. We will use a permeability field defined on a 50×50 grid that was generated by GSLIB [**16**]. It is shown in figure 3.4.1. The plot is in a $\log_e$ scale so the permeability varies over 3 orders of

FIGURE 3.4.2. Numerical diffusion in the presence of fast channels



Cartesian (left) and pressure-streamline (right) computation snapshots on a 50×50 grid, using the permeability field in figure 3.4.1.

TABLE 7. L_1 error of fine pressure-streamline method

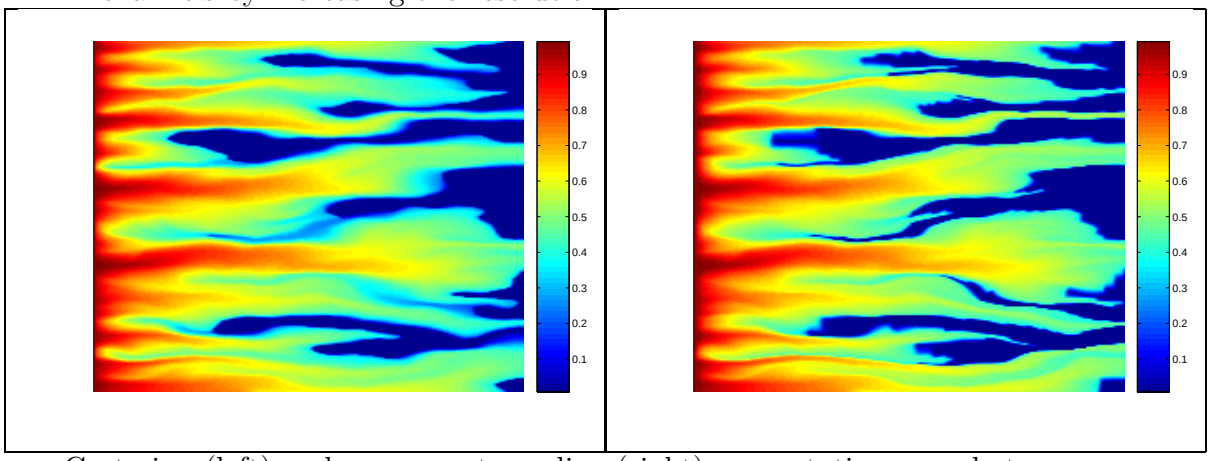
	50	100	200
Cartesian	0.165	0.100	0.0524
p, ψ	0.119	0.0616	0.0329

magnitude. It is strongly layered in the x-direction so we expect fast channels in the x-direction. We will solve the problem described in (2.1.16) with inlet at $x = 0$, no flow boundary conditions at $y = 0, 1$, and $m = 5$. The saturation profiles appear for time $t = 0.4$ in figure 3.4.2.

It is clear that near the shock the Cartesian method has more diffusion, especially in the cross-wind direction. The pressure-streamline method resolves accurately the fast channels but is slightly less accurate in the slow regions, compared to the Cartesian method. A comparison with a more resolved computation on a 200×200 grid in figure 3.4.3 confirms this.

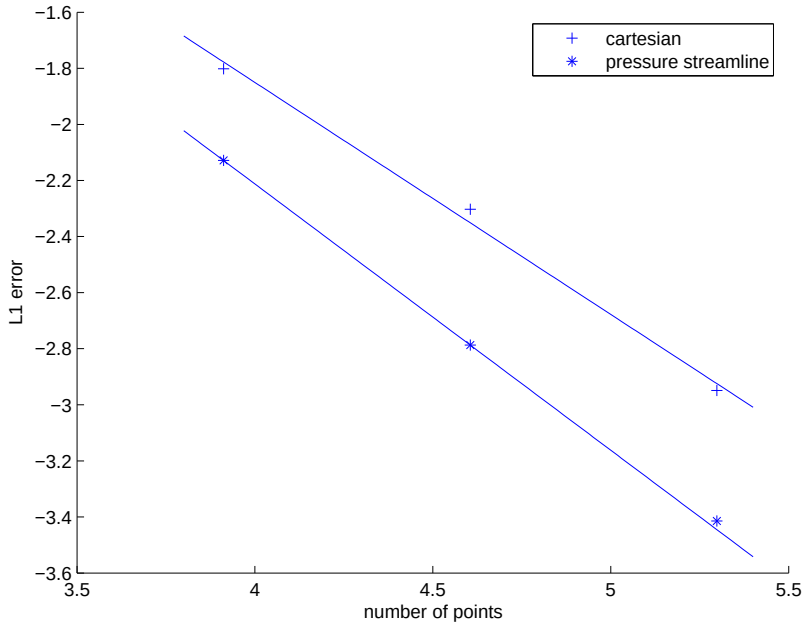
To investigate the convergence properties of the two methods we compute the L_1 norm of the error. For computations on grids finer than 50×50 we interpolated the permeability of figure 3.4.1 linearly. We considered a computation with 800×800 to be the exact solution. It is shown in table 7. Both schemes converge to first order because the solution is discontinuous. The pressure-streamline method converges faster because there it has less numerical diffusion. The performance of the two numerical schemes will vary even more for rougher permeability fields and flows in a

FIGURE 3.4.3. Decreasing numerical diffusion in the presence of fast channels by increasing the resolution



Cartesian (left) and pressure-streamline (right) computation snapshots on a 200×200 grid, using the permeability field in figure 3.4.1.

FIGURE 3.4.4. Convergence rate of the fine Cartesian and pressure-streamline methods



Log-log plot of the L_1 errors as functions of the number of discretization points and their least squares lines.

45 degree angle, due to cross-wind diffusion. For both schemes the L_1 error decreases as Δx, but the constant for the pressure-streamline method is smaller. This becomes obvious in a log-log plot of the error in figure 3.4.4.

Part 2

Upscaled Scheme

We try to construct numerical methods when the grid does not resolve the fast scale ϵ, that is, when the grid size satisfies $h > \epsilon$. A straightforward discretization on such a grid would fail because the solution is not smooth over the grid cells and the discretization error, the higher-order terms in a Taylor expansion, is large. If we take the limit as $h, \epsilon \to 0$ keeping $h > \epsilon$ the numerical method would not be consistent. Therefore we need to find an upscaled equation and design a numerical method for it instead.

Chapter 4

Upscaling One-Phase Flow

4.1. Two-Scale Limit

4.1.1. Context of the Present Work. In this chapter we investigate the properties of the saturation when it is convected by a velocity field that has $O(1)$ variation over length scales of $O(\epsilon)$. Up to a certain length scale the evolution of the saturation equation is dominated by the diffusion due to capillary forces, and upscaling the full two-phase flow equation consists of simply upscaling the pressure equation [**2**]. For velocity variation over larger length scales when the saturation equation is convection dominated its upscaling is no longer trivial. This is why we neglected diffusion altogether in our model equations.

We would like to average the saturation over all length scales, but keep information on the scale ϵ^{-1} intact, which is equivalent to taking the two-scale limit, introduced by Nguetseng [**41**]. We will focus on (3.2.2) without the terms due to the motion of the coordinate frame $p(x, y, t), \psi(x, y, t)$. This is the case of one-phase flow where the pressure is not coupled to the saturation and therefore time-independent, or the case of two-phase flow with an IMPES discretization, as has been described before. Then, in a Cartesian frame we have

$$\begin{aligned} S_t^\epsilon + \mathbf{v}^\epsilon \cdot \nabla S^\epsilon &= 0 \\ S(x, y, t = 0) &= S_{IC} \end{aligned} \tag{4.1.1}$$

and in the pressure-streamline frame we have

$$\begin{aligned} S_t^\epsilon + v_0^\epsilon S_p^\epsilon &= 0 \\ S(p,\psi,t=0) &= S_{IC}(p,\psi,\tfrac{\psi}{\epsilon}), \end{aligned} \tag{4.1.2}$$

with $v_0^\epsilon = \frac{\|\mathbf{v}^\epsilon\|^2}{\lambda K}$. We have deliberately omitted from these equations the precise nature of the dependence of the velocity field on the slow and fast variables because it will be part of the discussion that follows.

Let $y = \frac{x}{\epsilon}$ denote the fast variables. If one assumes a two-scale expansion for (4.1.1) in (x, y) in a periodic domain, the expansion will not be unique, in general. A two-scale expansion with two independent variables contains more information than what can be determined by its equations for all orders. Hou, Xin [**29**] specified the conditions on the velocity field under which this expansion is unique. In their problem the assumption of periodic boundary conditions was used for the ergodic theory results. To arrive at a unique two-scale expansion, E [**19**] used the fact that the ϵ^{-1} equation for non-oscillatory initial conditions, $\mathbf{v} \cdot \nabla_y S = 0$, imposes that the two-scale limit be constant along the flow lines. He incorporated this piece of information in the two-scale expansion by modifying the definition of two-scale convergence. He thus restricted how much information the expansion contains and obtained a unique two-scale limit. However the equations by which the two-scale limit is defined don't offer any insight into its structure. Moreover there is no clear way to design an efficient numerical method based on them.

Consider the definitions of the weak and two-scale limit of a sequence of functions S^ϵ

$$\int S^\epsilon(x)\phi(x)dx \to \int S(x)\phi(x)dx, \forall\phi \tag{4.1.3}$$

$$\int S^\epsilon(x)\phi(x,\frac{x}{\epsilon})dx \to \int S_{full}(x,y)\phi(x,y)dxdy, \forall\phi \tag{4.1.4}$$

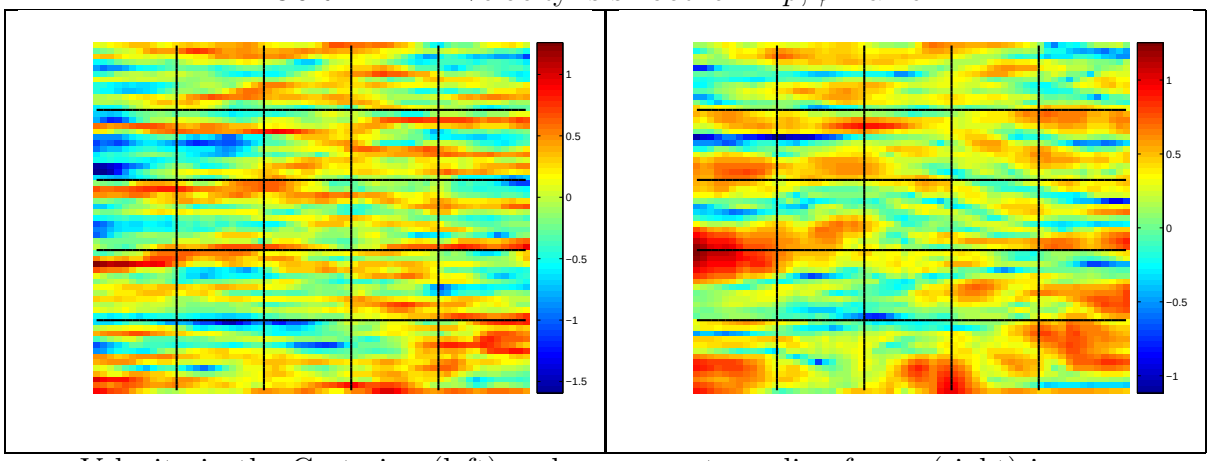
$$\int S^\epsilon(x)\phi(x,\frac{x}{\epsilon})dx \to \int S_{restricted}(x,y)\phi(x,y)dxdy, \quad \forall\phi \text{ with } \mathbf{v}\cdot\nabla_y\phi(x,y) = 0. \tag{4.1.5}$$

By setting $\phi(x,y) = \phi(x)$ in (4.1.4) and comparing with (4.1.3) we find $S(x) = \int S_{full}(x,y)dy$. Generalizing this argument we see that the weak limit is the average over all scales that are faster than the $O(1)$ of the corresponding two-scale limits. This shows that $S_{restricted}$ is the average of S_{full} along the fast streamlines, but contains fast scale information across the streamlines. Since we know from the ϵ^{-1} equation that the two-scale limit S_{full} is constant along the fast streamlines, the two are the same. The restricted sense of convergence has retained all information of the full two-scale limit.

Westhead [**53**] introduced an operator to project the full two-scale limit S_{full} onto its average along the streamlines that is the restricted $S_{restricted}$, thus obtaining a unique expansion. Projecting on the streamlines and restricting the test functions are equivalent. However their method differs from ours in that they first expanded the saturation equation into its moments and used the projection operator only to close the equation for the fluctuations. We applied it to the whole equation, by first deriving the two-scale limit using test functions. In both cases the equation for the average saturation contains a convection term and forcing by the fine scale. The advantage of our approach is that the convection term describes more accurately the mean flow, and thus less information is stored in the forcing term. For example in the case of linear flux, in our average saturation equation the convection term will give an accurate total flux, and the forcing from the small scales will only correct the breakthrough times of the fast channels. In addition our forcing term has a straightforward interpretation as the macrodispersion, which has been studied extensively.

The approaches of E and Westhead have in common that they assume a two-scale structure for the velocity field in the Cartesian variables. In most physical models, the assumption of periodic cells and fast variables in a Cartesian frame does not result from a physical property of the system. In our case the permeability will not in general have periodic cells in a Cartesian frame. We will assume a fast variable in the pressure-streamline frame. This idea appeared first in McLaughlin, Papanicolaou, and Pironneau [**40**] in the context of the Euler equations. The connectivity of the flow channels that results from the features of the permeability will be accurately reflected first in the velocity and then in the pressure and streamfunction variables.

FIGURE 4.1.1. Velocity is smoother in p, ψ frame



Velocity in the Cartesian (left) and pressure-streamline frame (right) in a logarithmic scale.

With a slight abuse of terminology, upscaling means averaging over cells, and we feel it is more intuitive to pick the cells over which we average to be regions between streamlines and lines of constant pressure.

To obtain more insight into our upscaling philosophy consider how the fast flow channels are upscaled with a Cartesian and a pressure-streamline method. In the Cartesian frame they are thin, and they are grouped together with slow channels resulting in a large upscaling error. In the pressure-streamline frame they are wider because the transformation focuses the computational points in the fast regions, and they are grouped separately from the slow channels. To demonstrate this we use the permeability in figure 4.1.1, which has many fast channels, and show the velocity $\frac{|\mathbf{v}|^2}{\lambda K}$ in the Cartesian and pressure-streamline frames. An additional advantage of selecting the fast variables in the pressure-streamline frame is that we can prove convergence of the upscaled equations without assuming either scale separation or periodic boundary conditions. This allows us to get rid of the awkward assumptions that the medium does not vary in some length scales and that when fluctuations exit the cell in one side they return to the opposite side of the same cell.

It is interesting to look at the structure of our multiscale expansion. If we write explicitly the dependent variables of the saturation we get

$$S^\epsilon(p,\psi) = \tilde{S}(p^\epsilon(x,y), \psi^\epsilon(x,y), \frac{p^\epsilon(x,y)}{\epsilon}, \frac{\psi^\epsilon(x,y,t)}{\epsilon}) + O(\epsilon).$$

In this work we assumed periodic cells in $\frac{p^\epsilon}{\epsilon}, \frac{\psi^\epsilon}{\epsilon}$, but it would have been possible to expand first the pressure and streamfuntion variables and look for the lowest-order term in the form $\tilde{S}(p_0(x,y), \psi_0(x,y), \frac{p_0(x,y)}{\epsilon}, \frac{\psi_0(x,y,t)}{\epsilon})$. We highlight the fact that the variables in this last expression no longer depend on ϵ. Physically this would mean that the multiscale structure is in terms of the lowest-order term for the pressure and not the full pressure. First Hou, Yang, and Wang [**30**] and later Hou, Yang, and Ran [**31**] implemented this idea for the Euler equations, assuming a two-scale structure in the variables of the Lagrangian map.

Even though the time of flight, streamfunction τ, ψ coordinate system has the advantages described above, it is not suitable for upscaling in our case. When we transform the equation to this frame there is no quantity with a fast scale; there is nothing to upscale. The oscillatory structure has been stored in the transformation. By using the pressure-streamline frame we choose to divide the oscillatory structure between the transformation and the velocity field and upscale the latter.

4.1.2. Derivation of the Two-Scale Limit for Linear Flux. We proceed with a rigorous derivation of the two-scale limit for a linear flux. The idea to consider the two-scale limit, due to Nguetseng [**41**], which is a weak limit using oscillatory test functions. To derive the two-scale limit we follow the method of E [**19**] to multiply the equation with an oscillatory test function, pass the derivatives to the test function and then take the limit $\epsilon \to 0$. In [**19**], E finds the two-scale limit for transport in conservation form assuming a multiscale structure in the Cartesian variables, whereas we will assume a multiscale structure in the pressure-streamline variables and deal with one-dimensional transport in nonconservative form. We denote the fast spatial variables by $\xi = \frac{p}{\epsilon}, \zeta = \frac{\psi}{\epsilon}$. We have suppressed the dependence of p on ϵ to simplify the notation, but the reader should keep it in mind as it has important implications on the structure of the asymptotic expansion. As we promised in the previous section, we assume that the velocity has the following structure

$$v_0^\epsilon(p,\psi) = v_0(p,\psi,\frac{p}{\epsilon},\frac{\psi}{\epsilon}).$$

Let $Y = Y_1 \times Y_2 = [0,1]^2$, ϕ be a smooth function, periodic in the variable in Y, with compact support and $J = \{\phi : \mathbb{R}^2 \times \mathbb{R}^+ \times Y \to \mathbb{R}\}$.

We will use the following convergence theorem due to Nguetseng [**41**]:

THEOREM 4.1.1. *Let $\{S^\epsilon\}_{\epsilon>0}$ be a uniformly bounded sequence in $L^\infty_{loc}(\mathbb{R}^+, L^2_{loc}(\mathbb{R}^2))$. Then there exists a subsequence, still denoted by $\{S^\epsilon\}_{\epsilon>0}$, and a function $\tilde{S} \in L^2_{loc}(\mathbb{R}^2 \times \mathbb{R}^+ \times Y)$, such that, as $\epsilon \to 0$*

$$\int_{\mathbb{R}^2\times\mathbb{R}^+} S^\epsilon(p,\psi,t)\phi(p,\psi,\frac{p}{\epsilon},\frac{\psi}{\epsilon},t)dpd\psi dt \to \int_{\mathbb{R}^2\times Y\times\mathbb{R}^+} \tilde{S}(p,\psi,\xi,\zeta,t)\phi(p,\psi,\xi,\zeta,t)dpd\psi d\xi d\zeta dt,$$

for all $\phi \in J$.

REMARK 4.1.2. We can give an indication why the two-scale limit of the saturation will not depend on a fast time variable if the initial condition $S_{IC}(p,\psi,\zeta)$ does not depend on $\frac{p}{\epsilon}$. If we assume a two-scale expansion where the first term depends on a fast time $\tau = \frac{t}{\epsilon}$ then the ϵ^{-1} equation will be

$$\begin{aligned} \tilde{S}_\tau + v_0\tilde{S}_{\frac{p}{\epsilon}} &= 0 \\ \tilde{S}(p,\psi,\xi,\zeta,t=0,\tau=0) &= S_{IC}(p,\psi,\zeta). \end{aligned}$$

The solution of this equation (with periodic boundary conditions) is $\tilde{S}(p,\psi,\xi,\zeta,t=0,\tau) = S_{IC}(p,\psi,\zeta)$, which shows that there is no fast time dependence in $\tilde{S}$ for $t=0$. We remind the reader that the structure of the asymptotic series is an assumption that either leads to a cascade of well-posed equations or not and that there does not need to be a justification for it.

To show that the sequence S^ϵ is bounded we multiply (4.1.2) by $\frac{S^\epsilon}{v_0^\epsilon}$ and integrate over the whole domain

$$\frac{d\|\frac{1}{v_0^\epsilon}S^\epsilon(\cdot,t)\|_2^2}{dt} = \int_{\mathbb{R}^2} S^\epsilon S^\epsilon_p dpd\psi = \int_{\mathbb{R}^2}((S^\epsilon)^2)_p dpd\psi = 0 \Rightarrow \|\frac{1}{v_0^\epsilon}S^\epsilon(\cdot,t)\|_2^2 \le \|\frac{1}{v_0^\epsilon}S^\epsilon(\cdot,0)\|_2^2.$$

v_0^ϵ is the Jacobian of a nonsingular transformation, and is therefore always positive. Assuming that it is bounded below boundedness of S^ϵ follows immediately. The conditions of Nguetseng's theorem apply so the two-scale limit $\tilde{S}$ exists.

To derive the two-scale limit we divide both sides of the equation (4.1.2) by the Jacobian, multiply by a test function ϕ^ϵ, and integrate

$$\int_{\mathbb{R}^2\times\mathbb{R}^+}(\frac{S^\epsilon_t}{v^\epsilon_0}\phi^\epsilon + S^\epsilon_p\phi^\epsilon)dpd\psi dt = 0.$$

We pass the derivatives in t and p to the test function by integration by parts and obtain

$$\int_{\mathbb{R}^2\times\mathbb{R}^+}(\frac{S^\epsilon}{v^\epsilon_0}\phi^\epsilon_t + S^\epsilon\phi^\epsilon_p)dpd\psi dt + \int_{\mathbb{R}^2}\frac{S_{IC}(p,\psi,\zeta)}{v^\epsilon_0(p,\psi)}\phi^\epsilon(p,\psi,t=0)dpd\psi = 0. \tag{4.1.6}$$

The boundary terms vanish because S^ϵ vanishes at ∞. This equation will be the starting point to derive equations for the two-scale limit to orders ϵ^{-1} and ϵ^0, as follows. First we choose $\phi^\epsilon = \epsilon\phi(p,\psi,\frac{p}{\epsilon},\frac{\psi}{\epsilon},t)$ in (4.1.6). With this substitution we expand the second term inside the first integral in (4.1.6)

$$\int_{\mathbb{R}^2\times\mathbb{R}^+}S^\epsilon\phi^\epsilon_p dpd\psi dt = \int_{\mathbb{R}^2\times\mathbb{R}^+}\epsilon S^\epsilon\phi_p dpd\psi dt + \int_{\mathbb{R}^2\times\mathbb{R}^+}S^\epsilon\phi_\xi(p,\psi,\frac{p}{\epsilon},\frac{\psi}{\epsilon},t)dpd\psi dt.$$

We take the limit $\epsilon\to 0$ in (4.1.6) as described in Theorem 4.1.3. Since the two-scale limit $\tilde{S}$ exists, S^ϵ will converge to it, with ϕ^ϵ_p, ϕ^ϵ_t, and v^ϵ_0 playing the role of the test functions of the theorem. As $\epsilon\to 0$ all terms vanish because they contain S^ϵ and ϕ, which are bounded and are premultiplied by ϵ, except the second term above. The term that does not vanish gives the $O(\epsilon^{-1})$ equation after another integration by parts in the fast variable ξ

$$\int_{\mathbb{R}^2\times Y\times\mathbb{R}^+}\tilde{S}_\xi\phi dpd\psi d\xi d\zeta dt = 0. \tag{4.1.7}$$

The boundary terms that arise from the integration by parts canceled each other out because ϕ and $\tilde{S}$ are periodic in ξ. The resulting $O(\epsilon^{-1})$ equation implies that the two-scale limit of the saturation does not depend on the fast variable ξ along the streamlines.

Next we choose a test function $\phi^\epsilon = \phi(p,\psi,\frac{\psi}{\epsilon},t)$ that does not depend on ξ. Thus, we incorporate the information from the $O(\epsilon^{-1})$ equation into the $O(\epsilon^0)$ equation. With this choice of test function, taking in (4.1.6) again the limit $\epsilon\to 0$, we find

$$\int_{\mathbb{R}^2\times Y\times\mathbb{R}^+}(\frac{\tilde{S}}{v_0}\phi_t + \tilde{S}\phi_p)dpd\psi d\xi d\zeta dt + \int_{\mathbb{R}^2\times Y}\frac{\tilde{S}_{IC}(p,\psi,\zeta)}{v_0(p,\psi,\xi,\zeta)}\phi(p,\psi,\zeta,t=0)dpd\psi d\xi d\zeta = 0.$$

We have denoted with $\tilde{S}_{IC}$ the two-scale limit of the initial condition. We will manipulate this equation a little farther by carrying the ξ integration out. We remind the reader that $(\xi, \zeta) \in Y = Y_1 \times Y_2$, as was defined previously.

$$\int_{\mathbb{R}^2\times Y\times\mathbb{R}^+}(\frac{\tilde{S}}{v_0}\phi_t+\tilde{S}\phi_p)d\xi dpd\psi d\zeta dt+\int_{\mathbb{R}^2\times Y}\frac{\tilde{S}_{IC}(p,\psi,\zeta)}{v_0(p,\psi,\xi,\zeta)}\phi(p,\psi,\zeta,t=0)dpd\psi d\xi d\zeta =$$
$$=\int_{\mathbb{R}^2\times Y_2\times\mathbb{R}^+}(\tilde{S}\int_{Y_1}\frac{d\xi}{v_0}\phi_t+\tilde{S}\phi_p)dpd\psi d\zeta dt+$$
$$+\int_{\mathbb{R}^2\times Y_2}\int_{Y_1}\frac{\tilde{S}_{IC}(p,\psi,\zeta)}{v_0(p,\psi,\xi,\zeta)}\phi(p,\psi,\zeta,t=0)d\xi dpd\psi d\zeta.$$

After another integration by parts in t and p for the first term we find

$$\int_{\mathbb{R}^2\times Y_2\times\mathbb{R}^+}(\tilde{S}_t\int_{Y_1}\frac{d\xi}{v_0}+\tilde{S}_p)\phi dpd\psi d\zeta dt+ \tag{4.1.8}$$
$$+\int_{\mathbb{R}^2\times Y_2}\tilde{S}(p,\psi,\zeta,t=0)\int_{Y_1}\frac{d\xi}{v_0}\phi(p,\psi,\zeta,t=0)dpd\psi d\xi d\zeta-$$
$$-\int_{\mathbb{R}^2\times Y_2}\int_{Y_1}\frac{\tilde{S}_{IC}(p,\psi,\zeta)}{v_0(p,\psi,\xi,\zeta)}d\xi\phi(p,\psi,\zeta,t=0)dpd\psi d\zeta=0.$$

The initial condition for $\tilde{S}$ must be $\tilde{S}(p,\psi,\zeta,t=0)=\tilde{S}_{IC}(p,\psi,\zeta)$ so that the second and third terms cancel. We define $\tilde{v_0}$ to be the harmonic average of the velocity along the fast streamline variable over a cell, $\tilde{v_0}=(\int_Y v_0^{-1}d\xi)^{-1}$, and rewrite the first term in terms of it

$$\int_{\mathbb{R}^2\times Y_2\times\mathbb{R}^+}(\int_{Y_1}(\frac{\tilde{S}_t}{v_0}+\tilde{S}_p)d\xi)\phi dpd\psi d\zeta dt=\int_{\mathbb{R}^2\times Y_1\times\mathbb{R}^+}(\frac{\tilde{S}_t}{\tilde{v_0}}+\tilde{S}_p)\phi dpd\psi d\zeta dt. \tag{4.1.9}$$

(4.1.7), (4.1.9) are the weak formulation of the equations for the two-scale limit of the saturation. The corresponding strong form is

$$\begin{aligned}\tilde{S}_\xi &= 0\\ \tilde{S}_t+\tilde{v_0}\tilde{S}_p &= 0\\ \tilde{S}(p,\psi,\zeta,t=0) &= \tilde{S}_{IC}(p,\psi,\zeta).\end{aligned} \tag{4.1.10}$$

The solution of these equations is unique because the system is linear.

We have just proved

THEOREM 4.1.3. *The solutions* $\{S^\epsilon\}_{\epsilon>0}$ *of (4.1.2) with* $v_0^\epsilon = v(p, \psi, \frac{p}{\epsilon}, \frac{\psi}{\epsilon})$ *2-scale converge to a unique limit* $\tilde{S}$ *given by (4.1.10), as* $\epsilon \to 0$.

We can consider the equations (4.1.10) to describe the homogenized operator at the expense of having an extra variable, namely the fast variable across the streamlines ζ. Later in this section we will show how to get rid of this extra variable. These equations have a physical interpretation that is very intuitive. Two points that start on the same streamline at the beginning of a cell and are traveling, one with the full velocity field and the other with the harmonic average of the velocity over the cell, will meet at the end of the cell. Therefore the harmonic average of the velocity is the correct average velocity by which the averaged saturation should travel. We will exploit this property in the next section, as well.

REMARK 4.1.4. The above results can be extended to the case when the initial condition depends on the fast scale ξ. The only difference in the two-scale limit will be the initial condition. From (4.1.8) it will be

$$\tilde{S}(p, \psi, \zeta, t = 0) = \left(\int_{Y_1} \frac{1}{v_0(p, \psi, \xi, \zeta)} d\xi \right)^{-1} \int_{Y_1} \frac{\tilde{S}_{IC}(p, \psi, \zeta)}{v_0(p, \psi, \xi, \zeta)} d\xi,$$

which imposes that the initial condition for $\tilde{S}$ be the average of $\tilde{S}_{IC}$ weighted by the velocity v_0.

4.1.3. Derivation of the Two-Scale Limit for Nonlinear Flux. We extend the results of the previous section to the saturation equation with nonlinear flux

$$\begin{aligned} S_t^\epsilon + v_0^\epsilon f(S^\epsilon)_p &= 0 \\ S(p, \psi, t = 0) &= S_0 \end{aligned} \tag{4.1.11}$$

where f is continuous on R.

The tools that we used for a linear flux are not powerful enough to prove convergence in the nonlinear case. If a sequence of functions u_j converges to u in L_2, this does not imply that $g(u_j)$ converges to $g(u)$ for a general continuous function g. We use Young measures to investigate the limit of $g(u_j)$. We will denote the Young

measure ν_x associated to the sequence $g(u_k)$ by $\langle \nu_x(\mu), g(\mu) \rangle$. To construct it, consider for fixed j the family of measures $\{\delta_{u_j(x)}\}$ parameterized by x, where $\delta_{u_j(x)}$ is a delta function centered at $(x, u_j(x))$ in the graph of $u_j(x)$. There is a one-to-one correspondence between the function $g(u_j)$, its graph, and its Young measure. Then ν_x is the limit of the convergent subsequence of this family. Intuitively, the Young measure at a point x is the limiting density of the values $u_j(x)$ at x. When the Young measure reduces to a delta function at x, then for that point we have indeed $g(u_j(x)) \to g(u(x))$. The formulation of theorem 4.1.3 with Young measures follows.

THEOREM 4.1.5. *Assume we have a sequence of functions* $\{S^\epsilon\}_{\epsilon>0}$ *in* L^∞_{loc} *with* $S^\epsilon : \mathbb{R}^+ \times \mathbb{R}^2 \to K$ *where* K *is a compact subset of* $\mathbb{R}$. *Then there exists a subsequence, still denoted by* $\{S^\epsilon\}_{\epsilon>0}$, *and a family of parameterized probability measures* $\{\nu_{p,\psi,\xi,\zeta,t}(\mu)\}$ *supported in* K, *which depends measurably on* (p, ψ, ξ, ζ, t) *and is periodic in* ξ, η *with period* $Y = [0,1]^2$ *such that as* $\epsilon \to 0$

$$\int_{\mathbb{R}^2 \times \mathbb{R}^+} f(S^\epsilon(p,\psi)\phi(p,\psi,\frac{p}{\epsilon},\frac{\psi}{\epsilon},t)dpd\psi dt \to \int_{\mathbb{R}^2 \times Y \times \mathbb{R}^+} \langle \nu_{p,\psi,\xi,\zeta,t}(\mu), f(\mu) \rangle \, \phi(p,\psi,\xi,\zeta,t)dpd\psi d\xi d\zeta dt,$$

for every nonnegative $\phi(p,\psi) \in J$, $f \in C(K)$.

This theorem is a simple extension of Nguytseng's theorem and appeared in E [**19**]. Following the methodology of E [**19**], we will first derive the entropy condition that the two-scale limit must obey and then use it to prove the convergence to the two-scale limit. Compared to the case with a linear flux, here we must tackle the fact that instead of an equality for S^ϵ, we must work with an inequality for S^ϵ that holds for all entropies. We consider the family of Kruzkov's entropies

$$\eta(\mu, k) = |\mu - k|, q(\mu, k) = sgn(\mu - k)(f(\mu) - f(k)).$$

If an entropy inequality is satisfied for all Kruzkov entropies then it is satisfied for any linear combination with positive coefficients of the Kruzkov entropies, that is, for any convex entropy, and this implies uniqueness of the weak solution. To consider all entropy functions is equivalent to considering Kruzkov's entropies. With

the family of Kruzkov entropies, the entropy inequality (3.2.6) for a time-independent transformation reduces to

$$\int_{\mathbb{R}^2\times\mathbb{R}^+} J^{-1}\eta(S^\epsilon,k)\phi_t+q(S^\epsilon,k)\phi_p dpd\psi dt+$$

$$\int_{\mathbb{R}^2} \eta(S(p,\psi,t=0))\phi(p,\psi,t=0)J^{-1}dpd\psi \geq 0. \tag{4.1.12}$$

As in the linear case, we will use this equation as a starting point to derive equations for the two-scale limit $\tilde{S}$ to $O(\epsilon^{-1})$ and $O(\epsilon^0)$ by substituting in it carefully selected test functions. Substituting first $\phi = \epsilon\phi(p,\psi,\frac{p}{\epsilon},\frac{\psi}{\epsilon},t)$ and taking the limit $\epsilon \to 0$ we find

$$\int_{\mathbb{R}^2\times Y\times\mathbb{R}^+} \langle \nu_{p,\psi,\xi,\zeta,t}(\mu), q(\mu,k)\rangle\, \phi_\xi dpd\psi d\xi d\zeta dt \geq 0.$$

In the linear case the $O(\epsilon^{-1})$ equation showed that $\tilde{S}$ did not depend on ξ and thus allowed us to make the test functions independent of ξ. To show that the equation of the two-scale limit will not depend on ξ, we must show that $q(\tilde{S})$ and $\eta(\tilde{S})$ do not depend on ξ, or more precisely, that the corresponding Young measures don't depend on ξ. We will first deal with $\langle \nu_{p,\psi,\xi,\zeta,t}(\mu), q(\mu,k)\rangle$. The above relation means that $\langle \nu_{p,\psi,\xi,\zeta,t}(\mu), q(\mu,k)\rangle_\xi$ is positive. For every ϵ, $q_\xi(S^\epsilon,k)$ has zero mean in ξ because it is periodic in that variable. Then, also its limit $\langle \nu_{p,\psi,\xi,\zeta,t}(\mu), q(\mu,k)\rangle_\xi$ must have zero mean in ξ. A nondecreasing function with zero mean must be zero. Then it must be

$$\int_{\mathbb{R}^2\times Y\times\mathbb{R}^+} \langle \nu_{p,\psi,\xi,\zeta,t}(\mu), q(\mu,k)\rangle\, \phi_\xi dpd\psi d\xi d\zeta dt = 0. \tag{4.1.13}$$

We have shown that $\langle \nu_{p,\psi,\xi,\zeta,t}(\mu), q(\mu,k)\rangle$ is independent of ξ. The entropy inequality (4.1.12) holds for any smooth entropy-entropy flux pair $\eta(\mu)$, $q = \int \eta'(\mu)f'(\mu)d\mu$ as can be easily verified by substitution. If $f' \neq 0$ then any smooth function can be written as $\int \eta'(\mu)f'(\mu)d\mu$ so (4.1.13) holds for any smooth q. Furthermore since C^∞ is dense in L^∞ and differentiation and integration are continuous operations, (4.1.13) holds for any $q \in L^\infty$. In particular

$$\int_{\mathbb{R}^2\times Y\times\mathbb{R}^+} \langle \nu_{p,\psi,\xi,\zeta,t}(\mu), \eta(\mu,k)\rangle\, \phi_\xi dpd\psi d\xi d\zeta dt = 0. \tag{4.1.14}$$

We have shown that the limits of $\eta(S^\epsilon,k), q(S^\epsilon,k)$ don't depend on ξ so we will remove that subscript from ν. Letting $\phi = \phi(p,\psi,\frac{\psi}{\epsilon},t)$ and taking the limit $\epsilon \to 0$ in

(4.1.12) we obtain

$$\int_{\mathbb{R}^2\times Y\times\mathbb{R}^+} J^{-1}\left\langle \nu_{p,\psi,\zeta,t}(\mu),\eta(\mu,k)\right\rangle \phi_t(p,\psi,\zeta,t)+ \\ +\left\langle \nu_{p,\psi,\zeta,t}(\mu),q(\mu,k)\right\rangle \phi_p(p,\psi,\zeta,t)dpd\psi d\xi d\zeta dt+ \\ +\int_{\mathbb{R}^2}\eta(S(p,\psi,t=0))\phi(p,\psi,t=0)J^{-1}dpd\psi\geq 0.$$

Passing the ξ integration inside the integrand

$$\int_{\mathbb{R}^2\times Y_1\times\mathbb{R}^+}\int_{Y_2} J^{-1}d\xi\left\langle \nu_{p,\psi,\zeta,t}(\mu),\eta(\mu,k)\right\rangle \phi_t(p,\psi,\zeta,t)+ \\ +\left\langle \nu_{p,\psi,\zeta,t}(\mu),q(\mu,k)\right\rangle \phi_p(p,\psi,\zeta,t)dpd\psi d\xi d\zeta dt+ \\ +\int_{\mathbb{R}^2}\eta(S(p,\psi,t=0))\phi(p,\psi,t=0)J^{-1}dpd\psi\geq 0, \tag{4.1.15}$$

we obtain the weak form of the entropy inequality for the two-scale limit. We focus for a moment on the term $\langle \nu_{p,\psi,\zeta,t}(\mu),\eta(\mu,k)\rangle$. It describes the limit of the family of functions$|S^\epsilon-k|$for all k that may depend on space. We will guess $\tilde{S}$ and substitute it for k. Then we will show that the resulting Young measure $\left\langle \nu_{p,\psi,\zeta,t}(\mu),\left|\mu-\tilde{S}\right|\right\rangle=\delta_{\tilde{S}}$, completing the proof. To obtain the guess we can follow the same procedure as in the linear case to obtain the following equations for nonlinear flux

$$\begin{aligned}\tilde{S}_\xi &= 0\\ \tilde{S}_t+\tilde{v_0}f(\tilde{S})_p &= 0.\end{aligned} \tag{4.1.16}$$

We will only give a formal argument for convergence here. The strong form of (4.1.15), with $k=\tilde{S}$ is

$$\int_{Y_2} J^{-1}d\xi\left\langle \nu_{p,\psi,\zeta,t}(\mu),\eta(\mu,\tilde{S})\right\rangle_t+\left\langle \nu_{p,\psi,\zeta,t}(\mu),q(\mu,k)\right\rangle_p \leq 0.$$

Integrating with respect to p,ψ the second term vanishes because of periodicity and we obtain

$$\frac{d}{dt}\int_{\mathbb{R}^2}\int_{Y_2} J^{-1}d\xi\left\langle \nu_{p,\psi,\zeta,t}(\mu),\left|\mu-\tilde{S}\right|\right\rangle dpd\psi \leq 0.$$

At $t = 0$ we have $\int \left\langle \nu_{p,\psi,\zeta,t}(\mu), \left|\mu - \tilde{S}\right| \right\rangle dpd\psi = 0$, so the solution to this ordinary differential equation is

$$\int_{\mathbb{R}^2} \int_{Y_2} J^{-1} d\xi \left\langle \nu_{p,\psi,\zeta}(\mu), \left|\mu - \tilde{S}\right| \right\rangle dpd\psi \;\; = \;\; 0,$$

which shows that ν is the Dirac measure $\delta_{\tilde{S}}$. This argument can be made rigorous with the work of DiPerna [**17**].

We summarize these results in theorem 4.1.6.

THEOREM 4.1.6. *The solutions $\{S^\epsilon\}_{\epsilon>0}$ of (4.1.11) with $v_0^\epsilon = v(p, \psi, \frac{p}{\epsilon}, \frac{\psi}{\epsilon})$ satisfying the entropy condition (4.1.12) converge to a unique limit $\tilde{S}$ given by (4.1.16) and subject to the entropy condition (4.1.15), as $\epsilon \to 0$.*

4.1.4. Convergence Rate to the Two-Scale Limit. We provide a convergence proof of the fine saturation S^ϵ to the two-scale limit $\tilde{S}$ as $\epsilon \to 0$, without the assumption of scale separation or periodicity of fast variables. We also want to have an idea of how fast this convergence is. In what follows, quantities depend on ψ, ζ parametrically but we will suppress that dependence. In this notation the velocity satisfies

$$v_0^\epsilon(p) = v_0(p, \frac{p}{\epsilon}).$$

We will prove

THEOREM 4.1.7. *Assume that $v_0^\epsilon(p)$ is bounded uniformly in ψ, ζ above and below*

$$C^{-1} \leq v_0^\epsilon(p) \leq D. \tag{4.1.17}$$

The solution $\tilde{S}$ of (4.1.10) converges to S defined by (4.1.2) with $v_0^\epsilon = v(p, \psi, \frac{p}{\epsilon}, \frac{\psi}{\epsilon})$ and initial conditions that may depend on the fast scale at a rate given by

$$\|S^\epsilon - \tilde{S}\|_\infty \leq G\epsilon,$$

when the initial conditions are Lipschitz, and

$$\|S^\epsilon - \tilde{S}\|_n \leq G\epsilon^{1/n},$$

when they have a finite number of discontinuities.

The velocity bound implies that $\tilde{C}^{-1} \leq \tilde{v_0}(p) \leq \tilde{D}$, uniformly in ψ, ζ . We will we look at the properties of the flow maps and then use these properties to obtain the convergence estimate. For a particle that starts at point p at $t = 0$ and moves with velocity v_0^ϵ, the flow map $P(p,T)$ is its position at time $t = T$. The flow maps $P(p,T), \tilde{P}(p,T)$ corresponding to $S, \tilde{S}$ are defined by

$$\begin{aligned} \tfrac{dP}{dT} &= v_0^\epsilon(P) \qquad & \tfrac{d\tilde{P}}{dT} &= \tilde{v_0}(P) \\ P(p,0) &= p \qquad & \tilde{P}(p,0) &= p. \end{aligned}$$

The velocities are given as functions of the spatial variable so a more useful function that characterizes particles motion under the velocity field is its time of flight T. $T(p,P)$ is the time required to travel between p and P. It is given by

$$\begin{aligned} \tfrac{dT}{dP} &= \tfrac{1}{v_0^\epsilon(P)} \qquad & \tfrac{d\tilde{T}}{dP} &= \tfrac{1}{\tilde{v_0}(P)} \\ T(p,p) &= 0 \qquad & \tilde{T}(p,p) &= 0. \end{aligned} \tag{4.1.18}$$

We can integrate these equations to find

$$T = \textstyle\int_p^P \frac{d\theta}{v_0^\epsilon(\theta)} \qquad \tilde{T} = \int_p^P \frac{d\theta}{\tilde{v_0}(\theta)} \ .$$

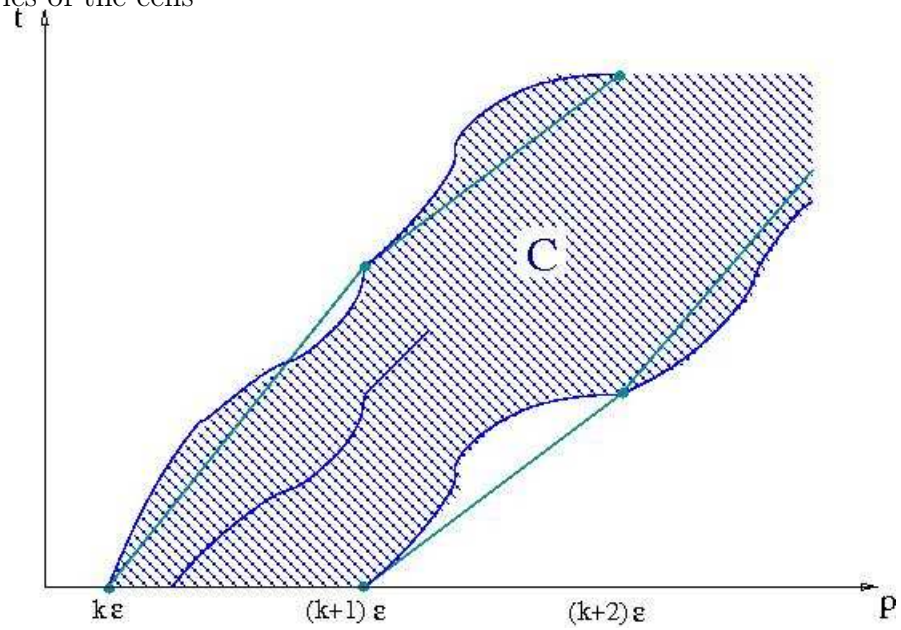
Using the fact that $\tilde{v_0}(P)$ is constant in a cell $(k\epsilon, (k+1)\epsilon)$ we can take it outside the integral

$$\begin{aligned} \tilde{T}(k\epsilon,(k+1)\epsilon) &= \int_{k\epsilon}^{(k+1)\epsilon} \frac{d\theta}{\tilde{v_0}(\theta)} = \frac{1}{\tilde{v_0}(k\epsilon+\frac{\epsilon}{2})} \int_{k\epsilon}^{(k+1)\epsilon} d\theta = \int_{k\epsilon}^{(k+1)\epsilon} \frac{d\theta}{v_0^\epsilon(\theta)} \\ &= T(k\epsilon,(k+1)\epsilon). \end{aligned}$$

Then $T, \tilde{T}$ agree when the starting point p and the ending point P are the boundaries of a cell. This is illustrated in figure 4.1.2. Time of flight is additive in the sense that $T(a,b) + T(b,c) = T(a,c)$, which implies the two time of flight surfaces agree on a grid of size ϵ in the (p,P) plane. For $p < k\epsilon < (k+q)\epsilon < P$, using the additivity of time of flight we get

$$\begin{aligned} |T(p,P) - \tilde{T}(p,P)| &\leq |T(p,k\epsilon) - \tilde{T}(p,k\epsilon)| \\ &+ |T(k\epsilon,(k+q)\epsilon) - \tilde{T}(k\epsilon,(k+q)\epsilon)| \\ &+ |T((k+q)\epsilon,P) - \tilde{T}((k+q)\epsilon,P)|. \end{aligned}$$

FIGURE 4.1.2. Fine and coarse characteristics intersect at the boundaries of the cells



A fine characteristic emanating between $k\epsilon$ and $(k+1)\epsilon$ must remain in C.

The second term of the right hand side vanishes as we showed before. Using the definition (4.1.18) and the bounds on the velocities of $T, \tilde{T}$ we find

(4.1.19)

$$|T(p,P) - \tilde{T}(p,P)| \leq |T(p,k\epsilon) - \tilde{T}(p,k\epsilon)| + |T((k+q)\epsilon,P) - \tilde{T}((k+q)\epsilon,P)| \leq 2C\epsilon.$$

We stress that this bound is uniform in ψ, that is, over all streamlines because we assumed a uniform bound for the velocity in (4.1.17). Using this bound on the time of flight (4.1.19), simple calculus gives a bound on the inverse flow map

$$|p(T,P) - \tilde{p}(T,P)| \leq \frac{1}{\min_{p,\psi} \tilde{T}_p} |T(p,P) - \tilde{T}(p,P)| \leq 2C\tilde{C}\epsilon. \tag{4.1.20}$$

We will use this bound on the inverse flow to quantify the difference between the saturation and its two-scale limit. Now we will state the dependence on ψ explicitly. All norms are with respect to p and ψ. We can write the solution to the saturation equation and its two-scale limit using the initial condition and the inverse flow map as

$$S^\epsilon = S_0(p(T,P,\Psi),\Psi) \qquad \tilde{S} = S_0(\tilde{p}(T,P,\Psi),\Psi). \tag{4.1.21}$$

For Lipschitz initial conditions with constant M it follows immediately that $\|S^\epsilon - \tilde{S}\|_\infty \leq M\|p - \tilde{p}\|_\infty \leq N\epsilon$, for some constant N. If the initial condition is not Lipschitz we do not have pointwise convergence. Consider the case where the initial condition has one jump discontinuity and is Lipschitz everywhere else. Then at time T also S^ϵ will have a discontinuity, by (4.1.21). To fix the notation, let that discontinuity be of magnitude less than ΔS, which does not have to be small, along a curve $(P(\lambda), \Psi(\lambda)), \lambda \in (0,1)$ on the P, Ψ plane, with length L. We will denote the thin strip of width $2C\tilde{C}\epsilon$ around that curve with

$$A_1 = \{(P, \Psi) \text{ such that } P(\lambda) - 2C\tilde{C}\epsilon \leq P \leq P(\lambda) + 2C\tilde{C}\epsilon, \lambda \in (0,1)\}$$

and the rest of the domain will be A_2. We selected the width of the strip based on (4.1.20) so that for any point $P(p,T)$ outside the strip, when we trace it back with the fine and coarse flow maps, if p is on one side of the discontinuity then $\tilde{p}$ cannot be on the other side. In A_2, p and $\tilde{p}$ are always on the same side of the jump so we can use the Lipschitz condition on S_0 to show that $\int_{A_2}(S^\epsilon - \tilde{S})^2 dA_2 \leq M^2\epsilon^2|A_2|$. Inside the strip even though S^ϵ and $\tilde{S}$ differ by an $O(1)$ quantity, we can use the area of the strip to make the L_2 norm of their difference small, that is, $\int_{A_1}(S^\epsilon - \tilde{S})^2 dA_1 \leq (\Delta S + N\epsilon)^2 4C\tilde{C}\epsilon L$. Then $\|S^\epsilon - \tilde{S}\|_2 \leq G\epsilon^{1/2}$. The estimates for any L^p follow in a similar fashion.

The results can be extended to the case of a finite number of discontinuities in the same way, which completes the proof.

To obtain the estimate for the saturation equation with linear flux we followed a longer path than necessary, but one that gives an intuitive explanation well. A simpler way to obtain the estimate, which holds only for initial conditions that don't depend on the fast scale, is given next, for the saturation equation with nonlinear flux.

THEOREM 4.1.8. *Assume that $v_0^\epsilon(p)$ is bounded uniformly in ψ, ζ above and below*

$$C^{-1} \leq v_0^\epsilon(p) \leq D.$$

Denote by $F(t,T)$ the solution to $S_t + f(S)_T = 0$. The solution $\tilde{S}$ of (4.1.16) converges to S^ϵ defined by (4.1.2) with $v_0^\epsilon = v(p, \psi, \frac{p}{\epsilon}, \frac{\psi}{\epsilon})$ and initial conditions that don't depend on the fast scale, at a rate given by

$$\|S^\epsilon - \tilde{S}\|_\infty \leq G\epsilon,$$

when F remains Lipschitz for all time, and

$$\|S^\epsilon - \tilde{S}\|_n \leq G\epsilon^{1/n},$$

when F develops at most a finite number of discontinuities.

PROOF. As before the velocity bound implies that $\tilde{C}^{-1} \leq \tilde{v_0}(p) \leq \tilde{D}$, uniformly in ψ, ζ. We transform the equations for S^ϵ (4.1.11) and $\tilde{S}$ (4.1.16) to the time of flight variable defined by

$$\begin{array}{rclcrcl} \frac{dT^\epsilon}{dp} & = & \frac{1}{v_0^\epsilon(p,\psi)} & \text{for } S^\epsilon \quad \text{and} & \frac{d\tilde{T}}{dp} & = & \frac{1}{\tilde{v}(p,\psi,\frac{\psi}{\epsilon})} \quad \text{for } \tilde{S}. \\ T^\epsilon(0) & = & 0 & & \tilde{T}(0) & = & 0 \end{array}$$

Both equations reduce to

$$S_t + f(S)_T = 0.$$

The solution to this equation is $F(t,T)$. Since the initial condition does not depend on ϵ neither does F. Then $S = F(t, T^\epsilon(P,\Psi))$, $\tilde{S} = F(t, \tilde{T}(P,\Psi))$. Using these expressions for the saturation we can obtain the desired estimates by following the same steps as in the linear case. When F remains Lipschitz for all times we can easily obtain a pointwise estimate in terms of the Lipschitz constant M $\|S^\epsilon - \tilde{S}\|_\infty = \|F(t,T^\epsilon) - F(t,\tilde{T})\|_\infty \leq M\|T^\epsilon - \tilde{T}\|_\infty \leq G\epsilon$. Otherwise we will need the time of flight bound (4.1.19) that we derived for the linear flux that reduces here to

$$|T^\epsilon(P) - \tilde{T}(P)| \leq 2C\epsilon. \tag{4.1.22}$$

We will divide the domain in regions where F is Lipschitz with constant M in the second variable, denoted by A_2, and shock regions, denoted by A_1, and estimate the

difference of S^ϵ and $\tilde{S}$ in each region separately. To fix the notation, let that there be n discontinuities in $F(t,\cdot)$ of magnitude less than ΔF, which does not have to be small, at $\{T = T_i\}_{i=1,\ldots,n}$. We will denote the thin strips of width $2C\epsilon$ around the discontinuities with A_1

$$A_1 = \{T \text{ such that } |T - T_i| \leq 2C\epsilon, \text{ for some } i = 1, \ldots, n\}$$

and with A_2 its complement. We selected the width of the strip based on (4.1.22), so that for any point P, if $T^\epsilon(P) \notin A_1$, then $T^\epsilon(P)$ and $\tilde{T}(P)$ are on the same side of any jump T_i. When $T^\epsilon(P) \in A_2$, F is Lipschitz in the region between T^ϵ and $\tilde{T}$, and we can show

$$\begin{aligned} \int_{A_2}(S^\epsilon - \tilde{S})^2 dpd\psi &= \int_{A_2}(F(t,T^\epsilon) - F(t,\tilde{T}))^2 dpd\psi \leq M^2\|T^\epsilon - \tilde{T}\|_\infty^2 |T^\epsilon(A_2)^{-1}| \\ &\leq N^2\epsilon^2 |T^\epsilon(A_2)^{-1}|, \end{aligned}$$

where we used the time of flight bound (4.1.22). By $|T^\epsilon(A_2)^{-1}|$ we denoted the image of A_2 under the inverse of $T^\epsilon(P)$. Inside the strip A_1, even though S^ϵ and $\tilde{S}$ differ by an $O(1)$ quantity we can use the smallness of the area of the strip to make the L_2 norm of their difference small

$$\begin{aligned} \int_{A_1}(S^\epsilon - \tilde{S})^2 dpd\psi &= \int_{A_2}(F(t,T^\epsilon) - F(t,\tilde{T}))^2 dpd\psi \leq (\Delta S + N\epsilon)^2 |T^\epsilon(A_1)^{-1}| \\ &\leq (\Delta S + N\epsilon)^2 4CDn\epsilon. \end{aligned}$$

We estimated the area $|T^\epsilon(A_1)^{-1}|$ by using the definition of A_1 and the fact that the Jacobian of the transformation $T^\epsilon(P)^{-1}$ is v_0^ϵ and is bounded uniformly in p, ψ as in equation (4.1.17). Putting together the two estimates for regions A_1 and A_2 we obtain $\|S^\epsilon - \tilde{S}\|_2 \leq G\epsilon^{1/2}$. Estimates in terms of the other L_p norms follow similarly. □

In general we are interested in obtaining a coarse scale method that gives accurate saturation profiles and fractional flow curves. If we relax this criterion and simply look for a coarse scale method that gives accurate fractional flow curves then the above proof shows that we only need to keep information across the streamlines in our coarse method. We can replace the velocity along each streamline with its geometric average along the whole streamline and solve the resulting equation on a coarse grid whose size does not depend on the magnitude of the velocity fluctuations. Then the effort to

solve the saturation equation is $O(1)$, and if the fine grid takes $O(N \times N)$ operations per time step, the above upscaled scheme will take $O(N \times 1)$ operations.

4.2. Comparison with the Cell Problem in the Cartesian Frame

We will rewrite the equations (4.1.10) in the form of an equation for the average saturation forced by a term that depends on the solution of a cell problem for the purposes of comparing our homogenized equations to those in the Cartesian frame. We expand $\tilde{S}$,$\tilde{v_0}$ as an average over the cells in the pressure-streamline frame and the corresponding fluctuations

$$\begin{aligned} \tilde{S} &= \overline{S}(p,\psi,t) + S'(p,\psi,\zeta,t) \\ \tilde{v_0} &= \overline{\tilde{v_0}}(p,\psi,t) + \tilde{v_0}'(p,\psi,\zeta,t). \end{aligned} \tag{4.2.1}$$

Averaging equations (4.1.10) with respect to ψ we find an equation for the mean of the saturation

$$\overline{S}_t + \overline{\tilde{v_0}\overline{S}_p} + \overline{\tilde{v_0}'S'_p} = 0.$$

An equation for the fluctuations is obtained by subtracting the above equation from (4.1.10)

$$S'_t + (\tilde{v_0} - \overline{\tilde{v_0}})\overline{S}_p + \tilde{v_0}S'_p - \overline{\tilde{v_0}'S'_p} = 0.$$

Together, the equations for the saturation are

$$\begin{aligned} \overline{S}_t + \overline{\tilde{v_0}\overline{S}_p} + \overline{\tilde{v_0}'S'_p} &= 0 \\ S'_t + \tilde{v_0}'\overline{S}_p + \tilde{v_0}S'_p - \overline{\tilde{v_0}'S'_p} &= 0. \end{aligned} \tag{4.2.2}$$

We can consider the second equation to be the cell problem and the first equation to be the upscaled equation. We remind the reader that the cell problem for a hyperbolic equation is $O(1)$ whereas for an elliptic equation it is $O(\epsilon)$. We can obtain an approximate numerical method by solving the cell problem only near the shock region in space time, where the macrodispersion term is largest. It is best to diagonalize these equations by adding the first to the second one

$$\begin{aligned} \overline{S}_t + \overline{\tilde{v_0}\overline{S}_p} &= -\overline{\tilde{v_0}'(\tilde{S}_p - \overline{S}_p)} \\ \tilde{S}_t + \tilde{v_0}\tilde{S}_p &= 0. \end{aligned} \tag{4.2.3}$$

Compared to (4.2.2), it has fewer forcing terms and no cross fluxes, which leads to a numerical method with less numerical diffusion that is easier to implement.

We want to compare the homogenized equations derived above with the corresponding equations in the Cartesian variables. We note that strictly speaking such a comparison is meaningless because the two homogenized equations correspond to different problems. Equations (4.2.3) are valid only when the coarse cells are defined by the level sets of pressure and the streamfunction whereas the homogenized equations in the Cartesian variables are valid under the assumption of x, y as the fast variables and periodically fluctuating velocities. In practical applications both of these assumptions become approximations to reality and then such a comparison is useful.

The homogenized equations in the Cartesian variables as derived by Westhead [**53**] are defined in terms of the average saturation over the coarse blocks $\overline{S}$ and the fluctuations S''. Note that whereas the fluctuations S' in the pressure-streamline frame depend only on one fast variable, the fluctuations S'' in the Cartesian frame depend on two fast variables. $\mathcal{P}$ is a projection operator onto the average along the streamlines within the cell, which corresponds to the fast variable along the streamlines and $\mathcal{Q}$ is a projection onto the orthogonal complement so that any function u can be written as $u = \mathcal{P}(u) + \mathcal{Q}(u)$. With this notation the homogenized equations are

$$\begin{aligned} \overline{S}_t + \overline{\mathbf{v}} \cdot \nabla \overline{S} + \nabla \cdot \overline{\mathbf{v}'' S''} &= 0 \\ S''_t + (\overline{\mathbf{v}} + \mathcal{P}(\mathbf{v}'')) \cdot \nabla S'' + \mathcal{P}(\mathbf{v}'') \cdot \nabla \overline{S} - \nabla \cdot \overline{\mathbf{v}'' S''} &= G(x, \tfrac{x}{\epsilon}, t), \end{aligned} \tag{4.2.4}$$

where

$$G(x, \frac{x}{\epsilon}, t) = (\overline{\mathbf{v}} + \mathcal{P}(\mathbf{v}'')) \cdot \mathcal{Q}(\nabla S'') - \mathcal{P}(\mathcal{Q}(\mathbf{v}'') \cdot \mathcal{Q}(\nabla S'')) + \mathcal{Q}(S'').$$

The Cartesian cell problem, which is the equation for the fluctuations in (4.2.4), is a two dimensional equation along two fast variables. Before one can solve it, one must compute the projections $\mathcal{P}$ and $\mathcal{Q}$ which adds to the complexity of the method and its computational cost. In contrast the pressure-streamline cell problem in equations

(4.2.3) contains only one fast variable and no projection operator. In some sense, in the pressure-streamline frame the projection operation, which was carried out by restricting the oscillatory test functions, cleanly removed a fast variable and reduced one fast dimension to arrive at the cell problem of (4.2.3). In the Cartesian frame the projection operation remained in the equations in the form of $\mathcal{P}$ and $\mathcal{Q}$ and the fast variation along the flow was not cleanly removed. This is another indication that the pressure-streamline frame reveals the structure of the flow correctly.

4.3. Weak Limit and Full Homogenization

With the derivation of the equation of the two-scale limit we have homogenized the fine saturation equations. The homogenized operator given by (4.1.16) still contains variation of order ϵ through the fast variable $\frac{\psi}{\epsilon}$; however there it does not contain any derivatives in that variable. Its dependence on $\frac{\psi}{\epsilon}$ is only parametric. We can consider $\frac{\partial}{\partial t} + \tilde{v}_0 \frac{\partial}{\partial p}$ to be the homogenized operator for $\frac{\partial}{\partial t} + v_0^\epsilon \frac{\partial}{\partial p}$ at the expense of having to introduce an extra dimension, an extra parameter.

We can remove the dependence of the homogenized operator on $\frac{\psi}{\epsilon}$ and arrive at a homogenized operator that is independent of the small scale. When we homogenized along the streamlines, the resulting equation was of hyperbolic type like the original equation. In a seminal and celebrated paper, Tartar [**49**] showed that homogenization across streamlines leads to transport with the average velocity plus a time-dependent diffusion term, referred to as macrodispersion, a physical phenomenon that was not present in the original fine equation. We briefly discuss this result here. It applies only in the case of linear flux.

We will assume that the velocity field does not depend on p inside the cells, that is, $v_0^\epsilon(p, \psi) = v_0(\psi, \frac{p}{\epsilon}, \frac{\psi}{\epsilon})$. With this assumption the equation for the two-scale limit of the saturation is

$$\tilde{S}(p, \psi, \frac{\psi}{\epsilon}, t)_t + \tilde{v}_0(\psi, \frac{\psi}{\epsilon}) \tilde{S}(p, \psi, \frac{\psi}{\epsilon}, t)_p = 0.$$

We will denote by L the Fourier transform in p and by F the Laplace transform in t. Taking these two transformations we find

$$LF\tilde{S}(q,\psi,\frac{\psi}{\epsilon},s)=\frac{FS_0(\psi,q)}{s+2\pi iq\tilde{v}_0(\psi,\frac{\psi}{\epsilon})}.$$

We denote by $d\nu_{\frac{\psi}{\epsilon}}$ the Young measure associated with the sequence $\tilde{v}_0(\psi,\cdot)$ and use it to obtain the weak limit $LF\overline{S}(q,\psi,s)$

$$LF\overline{S}(q,\psi,s)=FS_0(\psi,q)\int\frac{d\nu_{\frac{\psi}{\epsilon}}(\lambda)}{s+2\pi iq\lambda}=FS_0(\psi,q)\frac{1}{2\pi iq}\int\frac{d\nu_{\frac{\psi}{\epsilon}}(\lambda)}{\frac{s}{2\pi iq}+\lambda}. \tag{4.3.1}$$

To invert the Laplace and Fourier transforms we use the fact that there exists a second Young measure $d\mu_{\frac{\psi}{\epsilon}}$ that satisfies

$$\left(\int\frac{d\nu_{\frac{\psi}{\epsilon}}(\lambda)}{\frac{s}{2\pi iq}+\lambda}\right)^{-1}=\frac{s}{2\pi iq}+\overline{\tilde{v}}_0-\int\frac{d\mu_{\frac{\psi}{\epsilon}}(\lambda)}{\frac{s}{2\pi iq}+\lambda}.$$

We have denoted by $\overline{\tilde{v}}_0$ the weak limit of the velocity. If we insert this expression into (4.3.1), inverting the Laplace and Fourier transforms becomes straightforward. The resulting equation is

$$\overline{S}_t+\overline{\tilde{v}}_0\overline{S}_p=\int_0^t\int\overline{S}_{pp}(p-\lambda(t-\tau),\psi,\tau)d\mu_{\frac{\psi}{\epsilon}}(\lambda)d\tau. \tag{4.3.2}$$

This equation has no dependence on the small scale and we consider it to be the full homogenization of the fine saturation equation. Efendiev and Popov [**24**] have extended this method for the Riemann problem in the case of nonlinear flux.

4.4. Designing an Upscaled Model from the Two-Scale limit

It is hard to see how the fully homogenized equation (4.3.2), which we derived from the two-scale limit using Tartar's method, can be used to design a numerical method. We will derive an efficient numerical method starting from the two-scale limit by using the method of averaging. We will also demonstrate that this equation is very close to the homogenization result of the previous section. We will use the higher moments of the saturation and the velocity to model the macrodispersion. In the context of two-phase flow this idea was introduced by Efendiev, Durlofsky, and

Lee [**20, 21**] and is referred to as Volume Averaging. In these papers they were facing a much more difficult problem since they were averaging along and across the flow. In our case the velocity field is already upscaled in the direction along the flow. In addition we are selecting the shape of the cells so that the velocity varies less within each cell. The resulting algorithm is also simpler to implement in our case because of the decoupling of the full two-dimensional equation to one-dimensional transport along streamlines.

4.4.1. Physical Interpretation of the Subgrid Forcing. We would like a physical interpretation of the forcing term that appears in the equation for the mean saturation in (4.2.2). We will integrate the equation for the fluctuations along the characteristics to eventually form that forcing term. The characteristics are defined by

$$\frac{dP}{dt} = \tilde{v_0}, \text{ with } P(p,0) = p.$$

The equation for S' is

$$S' = -\int_0^t \left(\tilde{v_0}'(P(p,\tau),\psi)\overline{S}_p(P(p,\tau),\psi,\tau) + \overline{\tilde{v_0}'S_p'}) \right) d\tau.$$

We take the derivative of both sides with respect to p, multiply by $\tilde{v_0}'$, and average over ψ. The second term will be third- order in fluctuating quantities and therefore small compared to the first. The macrodispersion becomes

$$\overline{\tilde{v_0}'S_p'} = -\int_0^t \overline{\tilde{v_0}'\frac{\partial}{\partial p}\left(\tilde{v_0}'(P(p,\tau),\psi)\overline{S}_p(P(p,\tau),\psi,\tau)\right)}d\tau.$$

We take $\frac{\partial}{\partial p}$ outside the integral to find

$$\begin{aligned}\overline{\tilde{v_0}'S_p'} &= -\frac{\partial}{\partial p}\int_0^t \overline{\tilde{v_0}'\tilde{v_0}'(P(p,\tau),\psi)\overline{S}_p(P(p,\tau),\psi,\tau)}d\tau \\ &+ \int_0^t \overline{\frac{\partial \tilde{v_0}'}{\partial p}\tilde{v_0}'(P(p,\tau),\psi)\overline{S}_p(P(p,\tau),\psi,\tau)}d\tau.\end{aligned}$$

Looking at the last equation we can understand the influence of the macrodispersion in the upscaled equation (4.2.2). The first term corresponds to diffusion whose magnitude depends on the two point correlation of the velocity field, and the second corresponds to a convection. Similar results have been obtained in a probabilistic

framework (see for example [**43**]). When the velocity field does not vary a lot along the streamlines, which is realistic since we have already averaged along streamlines, we expect the convection term of the macrodispersion to be negligible. Convection by simple arithmetic average of the velocity across streamlines will then give the exact amount of fluid that is transported through the domain over large times. It will give the correct integral of the fractional flow curve over all time or the total amount of oil that can be obtained. If we want to retain some information about the breakthrough time or the precise form of the fractional flow rate at every time we have to retain some information about the fast and slow channels within the coarse cells. This information would be contained in the macrodispersion.

4.4.2. Numerical Averaging across Streamlines for Linear Flux. The derivation in the previous section contained no approximations, and the average saturation equation cannot be solved on the coarse grid. In this section we follow the same idea as in that derivation to solve the equation for the fluctuations along the characteristics, but with the purpose of deriving an equation on the coarse grid. To achieve our purpose we will not do analytical upscaling in the sense that we are not interested in deriving a continuous upscaled equation as in the previous section. We will first discretize the equation with a FV method in space and then upscale the resulting equation. Our upscaled equation is then dependent on the numerical scheme. We note that also the definitions of the average saturation and its fluctuations (4.2.1) are easier to understand in the context of a numerical scheme.

We use the same definition for the average saturation and the fluctuations as in (4.2.1) and follow the same steps until equation (4.2.2). We discretize the macrodispersion term in the equation for the average saturation

$$\overline{\tilde{v_0}' S_p'} = \frac{\overline{\tilde{v_0}' S'}^{i+1} - \overline{\tilde{v_0}' S'}^{i}}{\Delta p} + O(\Delta p).$$

A superscript $\cdot^i$ refers to a discrete quantity defined at the center of the conservation cell. Instead of solving the equation for the fluctuations on the fine characteristics as

before, which would lead to a fine grid algorithm, we solve it on the coarse characteristics defined by

$$\frac{dP}{dt} = \overline{\tilde{v_0}}, \text{ with } P(p,0) = p.$$

Compared to the equation that we obtained in the previous section for S', this equation for S' has an extra term, which appears second

$$S' = -\int_0^t \left(\tilde{v_0}'(P(p,\tau),\psi)\overline{S}_p(P(p,\tau),\psi,\tau) + \tilde{v_0}'(P(p,\tau),\psi)S'_p(P(p,\tau),\psi,\tau) + \overline{\tilde{v_0}'S'_p}) \right) d\tau.$$

The second term is second-order in fluctuating quantities, and we expect it to be smaller than the first term so we neglect it. As before, we multiply by $\tilde{v_0}'$ and average over ψ to find

$$\overline{\tilde{v_0}'S'} = -\int_0^t \overline{\tilde{v_0}'\tilde{v_0}(P(p,\tau),\psi)\overline{S}_p(P(p,\tau),\psi,\tau)} d\tau.$$

In this form at time t it is necessary to know information about the past saturation in $(0,t)$ to compute the future saturation. In the appendix we demonstrate that $\overline{S}_p(P(p,\tau)$ depends weakly on time, in the sense that the difference between $\overline{S}_p(P(p,\tau)$ and $\overline{S}_p(P(p,t)$ is of third-order in fluctuating quantities. Therefore we can take $\overline{S}_p(P(p,\tau)$ out of the time integral to find

$$\overline{\tilde{v_0}'S'} = -\int_0^t \overline{\tilde{v_0}'\tilde{v_0}'(P(p,\tau),\psi)} d\tau \overline{S}_p.$$

The term inside the time integral is the covariance of the velocity field along each streamline. The macrodispersion in this form can be computed independent of the past saturation.

4.4.3. Numerical Averaging across Streamlines for Nonlinear Flux. The nonlinearity of the flux function, the fact that the sum of the fine saturation fluxes is not equal to the flux of the sum of the fine saturations, introduces an extra source of error in the approximation. We Taylor expand $f(\tilde{S})$ near $\overline{S}$ and keep only the first

term

$$
\begin{aligned}
\tilde{S} &= \overline{S}(p,\psi,t) + S'(p,\psi,\zeta,t) \\
\tilde{v_0} &= \overline{\tilde{v_0}}(p,\psi,t) + \tilde{v_0}'(p,\psi,\eta,t) \\
\overline{f}(\tilde{S}) &= f(\overline{S}) + f_S(\overline{S})S' + O(S'^2) \\
f(S)_p &= f_S(\overline{S})\overline{S}_p + f(\overline{S})S' + \dots
\end{aligned} \tag{4.4.1}
$$

A reasonable objection is that this approximation will be inadequate near the shock since S' is not small there. The region near the shock is important because there the macrodispersion is largest. Due to the dependence of the jump in the saturation on the mobility we expect this approximation to be better for lower mobilities. Nevertheless this approximation works well in practice. For more accuracy it is also possible to retain more terms in the Taylor expansion, as was done in [**21**]. We will show that in realistic examples these higher-order terms are not important in our setting.

Using these definitions we derive the following equations for the average saturation and the fluctuations

$$
\begin{aligned}
\overline{S}_t + \overline{\tilde{v_0}} f(\overline{S})_p + \overline{\tilde{v_0}'(f_S(\overline{S})S')_p} &= 0 \\
S'_t + \tilde{v_0}' f_S(\overline{S})\overline{S}_p + \tilde{v_0} f_S(\overline{S})S'_p - \overline{\tilde{v_0}' S'_p} &= 0.
\end{aligned} \tag{4.4.2}
$$

The macrodispersion is discretized as

$$
\overline{\tilde{v_0}'(f_S(\overline{S})S')_p} = \frac{\overline{\tilde{v_0}' f_S(\overline{S})S'}^{i+1} - \overline{\tilde{v_0}' f_S(\overline{S})S'}^{i}}{\Delta p} + O(\Delta p).
$$

We solve the second equation on the coarse characteristics defined by

$$
\frac{dP}{dt} = \overline{\tilde{v_0}} f_S(\overline{S}), \text{ with } P(p,0) = p
$$

and form the terms that appear in the macrodispersion

$$
\overline{\tilde{v_0}' f_S(\overline{S})S'} = -\int_0^t \overline{\tilde{v_0}' f_S(\overline{S})\tilde{v_0}'(P(p,\tau),\psi) f_S(\overline{S}(P(p,\tau),\psi,\tau))\overline{S}_p(P(p,\tau),\psi,\tau)} d\tau.
$$

As before we have dropped terms that are second-order in fluctuating quantities. We use an argument in the appendix to show that $f_S(\overline{S}(P(p,\tau),\psi,\tau))\overline{S}_p(P(p,\tau),\psi,\tau)$

does not vary much along the streamlines and take it out of the integration in time to find

$$\overline{\tilde{v_0}' f_S(\overline{S}) S'} = -\int_0^t \overline{\tilde{v_0}' \tilde{v_0}'(P(p,\tau),\psi)} d\tau f_S(\overline{S})^2 \overline{S}_p. \tag{4.4.3}$$

This expression is very similar to the one obtained in the linear case; however here the macrodispersion depends on the past saturation through the equation for the coarse characteristics.

4.5. Implementation

4.5.1. Considerations for the Macrodispersion. Even though the macrodispersion depends on the past saturation it is possible to compute it incrementally. Given its value $D(t)$ at time t we compute the values at $t + \Delta t$ using the macrodispersion at the previous time

$$D(t+\Delta t) = \int_0^{t+\Delta t} \dots d\tau = \int_0^t \dots d\tau + \int_t^{t+\Delta t} \dots d\tau.$$

This is possible because in the derivation for the approximate expression for the macrodispersion we took the terms that depend on $S(\tau)$ outside the time integration. The integrand, the average covariance of the velocity field along the streamlines, needs to be computed only once at the beginning. Then updating the macrodispersion takes $O(n^2)$ computations, as many as it takes to update $\overline{S}$.

The macrodispersion can be negative because of the term v', and this leads to an equation with negative diffusion for which the numerical scheme is ill-posed. In [**21, 20**] an extra approximation was attempted to overcome this difficulty. We will simply apply the macrodispersion only where it is positive.

For the convection diffusion equation we must observe an extra CFL-like condition to obtain a stable numerical scheme [**47**]

$$\Delta t \leq \frac{\Delta p^2}{2\nu},$$

where ν is the diffusivity. In our case the diffusivity is $\int_{cell} \int_0^t \tilde{v_0}'(p(\tau),\psi)\tilde{v_0}'(p,\psi) d\tau d\psi$. If the macrodispersion is large this can be a very restrictive condition. We can use an implicit discretisation for the macrodispersion. This is straightforward since the

TABLE 1. Numerical demonstration of theorem 4.1.8

	2	4	8	16	32	64	128
L_1	0.0432	0.0124	0.0063	0.0049	0.0022	6.62×10^{-4}	1.41×10^{-4}
L_∞	0.670	0.664	0.663	0.665	0.653	0.61	0.096

problem is one-dimensional. The resulting system can be solved by a tridiagonal solver very fast. Since the order of the highest derivative in the equation has increased, we require extra boundary conditions. For the computation of the macrodispersion term, we impose no flux on both boundaries of the domain.

4.6. Numerical Results

To interpret correctly the numerical experiments that follow, we must distinguish between two sources of errors. We will refer to the difference between the upscaled and the exact equation as the upscaling or modeling error and to the difference between the solution of continuous upscaled equations and the solution to the numerical scheme as the discretization error. We will refer to the difference between the solutions of the continuous fine equations and the numerical scheme of the upscaled equations as the total error. To separate the upscaling error from the total error we will solve the upscaled equations on the fine grid, which is the grid on which we solve to the fine equation. We will also solve them on the coarse grid to compute the total error.

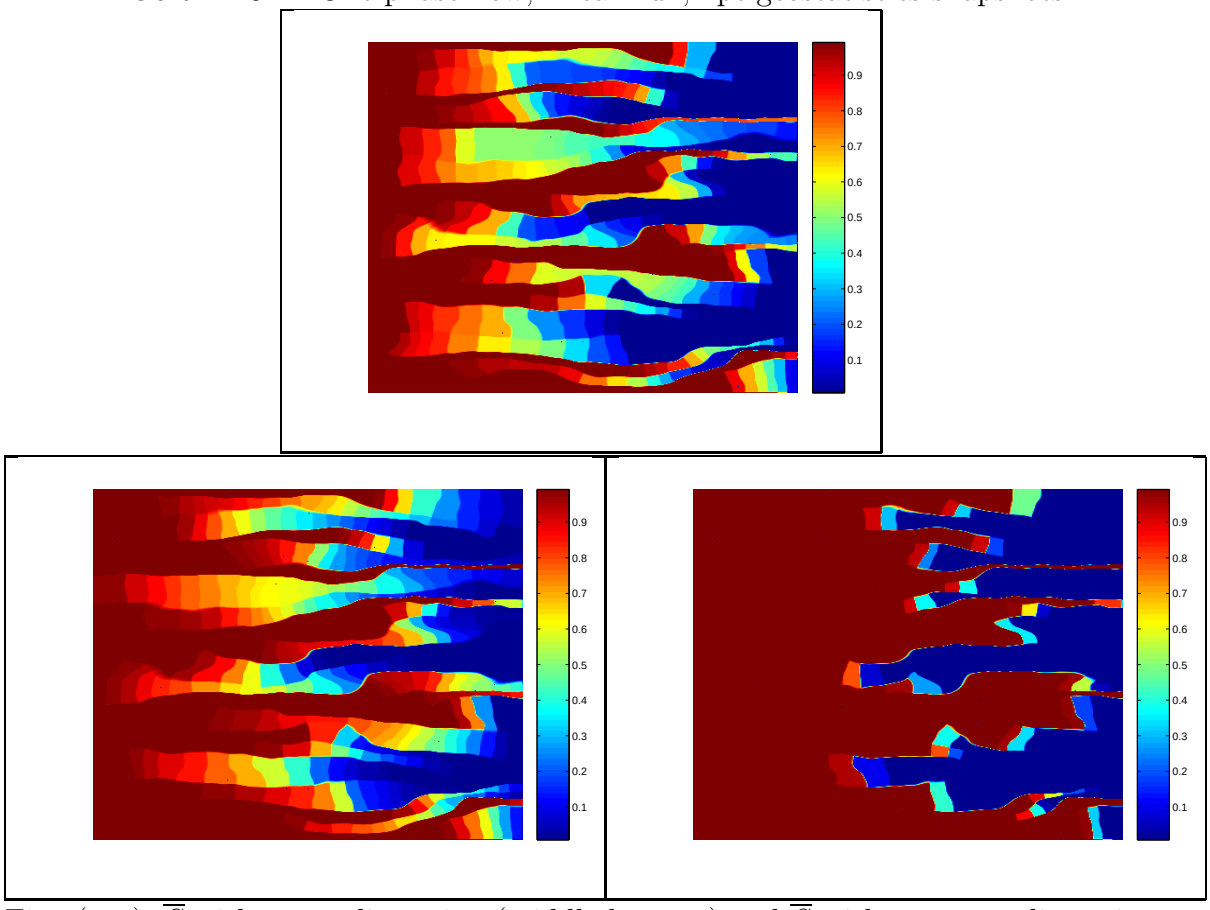
First we design a numerical experiment to demonstrate the estimate of theorem 4.1.8 for a discontinuous solution. To find the rate of convergence of $\tilde{S}$ to S^ϵ we have to use a grid that resolves the velocity and the shock so that the numerical diffusion near the shock does not mask the upscaling error. At the same time the velocity must vary enough in the cells so that the upscaling error is large. To avoid numerical diffusion we use a small final time. We will restrict ourselves to one spatial dimension, nonlinear flux with $m = 1$, $T_{final} = 0.1$, $v = 1 - 20p\sin(10\pi\frac{1}{p+0.1}) + 20sin(5\frac{\pi}{2})$. Since we are interested in the upscaling error we used 4096 cells for both upscaled and fine computation. The results are shown in table 1. The L_∞ norm shows that in all experiments with less than 64 points the numerical diffusion was not significant. The convergence rate seems to be slightly larger than 1, which is consistent with theorem 4.1.8.

In all the computations that follow we use permeability fields defined on 400×400 fine blocks. The fine computations are averaged over the coarse grid so that a comparison with coarse computations is possible. The coarse equations are computed on a fine grid to minimize the effects of numerical diffusion, except for the moving mesh computations that are computed on the coarse grid. We compare the saturation right before the breakthrough time so that the shock front is largest. This is the toughest case.

4.6.1. Macrodispersion Modeling. Before we show numerical results with the fully upscaled saturation we will demonstrate that the approximations involved in the derivation of the saturation equation with macrodispersion do not introduce large errors. When investigating the effects of macrodispersion numerically it is crucial that we minimize the numerical diffusion, otherwise the macrodispersion will be hidden by the numerical diffusion. If we Taylor expand the velocity v' in equation (4.4.3) then macrodispersion term scales as $t\Delta\psi^2$ for smooth velocity fields. If we use a Lax-Wendroff scheme with a limiter then in the shock region where the macrodispersion term is most important the numerical scheme is only first-order. The modified equation for Godunov's scheme shows that the numerical diffusion scales as $\overline{\tilde{v_0}}\Delta p(1-\overline{\tilde{v_0}}\frac{\Delta t}{\Delta p})$. Making sure that the numerical diffusion is smaller than the macrodispersion can be achieved in general by using coarse cells that are elongated in the ψ direction. Here we will simply discretize the equation for $\overline{S}$ on the fine grid.

We consider the case of linear flux to understand the influence of the macrodispersion and the accuracy of our model. We use the layered permeability field of figure 4.6.5 with no long-range correlation. We upscale from a 400×400 to a 50×50 grid. In table 4.6.1 we see snapshots of the saturation at $t = 0.3$. It is important to include the macrodispersion term when we are interested in accurate saturation profiles. In the computations of the next section we observe that the moving mesh that was implemented solves the equation for $\overline{S}$ with macrodispersion accurately even though the macrodispersion is computed on the adaptive grid, which is fine near the shock of $\overline{S}$ and coarse away from it. This confirms our conjecture that the macrodispersion is more significant near the shock. To derive the equation with the macrodispersion we

FIGURE 4.6.1. One-phase flow, linear flux, 2pt geostatistics snapshots



Fine (top), $\overline{S}$ with macrodispersion (middle bottom) and $\overline{S}$ without macrodispersion (right bottom).

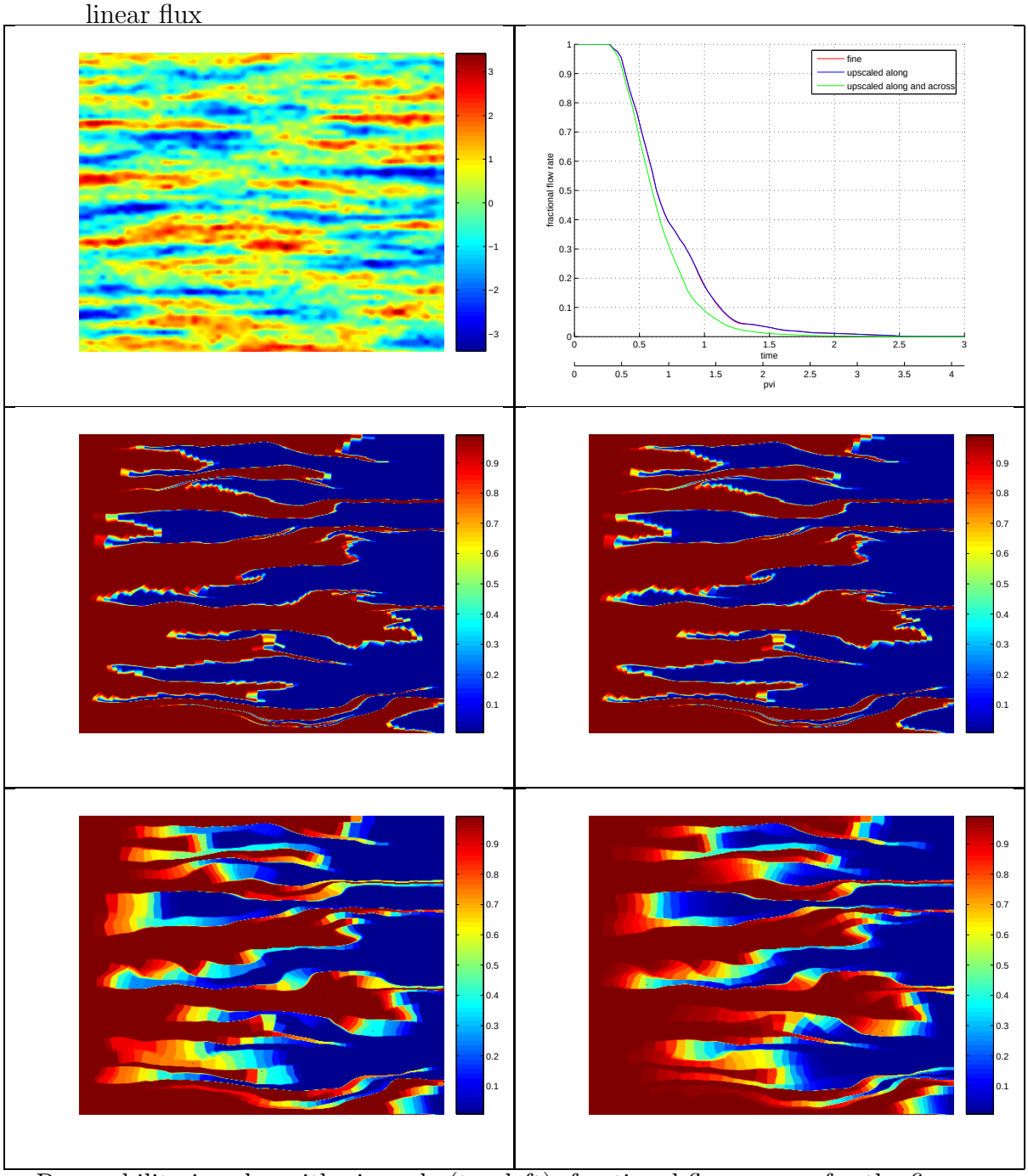
assumed that the velocity field varies little across the streamlines and we neglected higher-order terms. This explains the small discrepancy between the fine solution and the solution with macrodispersion.

4.6.2. Saturation Snapshots. We select three benchmark permeability fields, each representing a different class of permeabilities. The first is a layered permeability that we used so far where the velocity field varies rapidly across the flow but not along the flow and the flow is more or less parallel to the x-axis and was generated by GSLIB with $\frac{l_x}{l_y} = 10$. The second permeability is the Stanford 44 model [**10**] in the percolation limit where the flow has complicated geometrical features and has similar variation along and across the flow. The third permeability field is SPE10 36 [**11**] and has a fast channel that carries most of the flow. In the experiments with a nonlinear flux

we selected a mobility $m = 5$. The fine computation is on a 400×400 grid and is then averaged over the coarse grid, which is 50×50, for comparison. The numerical results, shown in figures 4.6.2, 4.6.3, 4.6.4, 4.6.5, 4.6.6, 4.6.7 show that the coordinate transformation captures the long-range correlations of the flow and the model with $\tilde{S}$ or $\overline{S}$ derived with homogenization and averaging provides an accurate upscaling model.

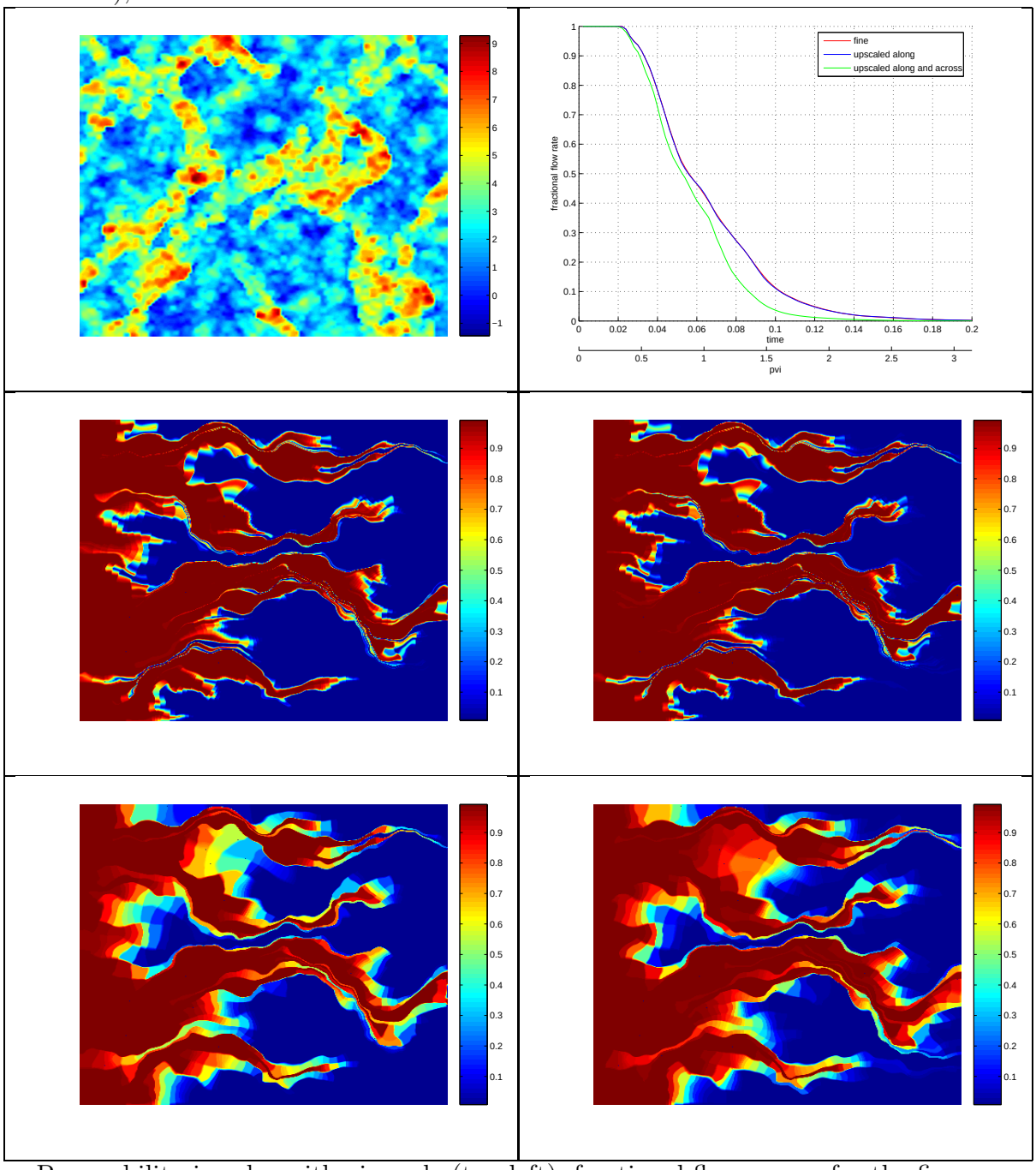
We notice that $\tilde{S}$ is much more accurate than $\overline{S}$, which makes sense because $\tilde{S}$ is upscaled only along the streamlines whereas $\overline{S}$ is upscaled along and across the streamlines. In the experiments with the fast channel, the SPE10 36 permeability field, we see an artifact next to the fast channel in the saturation snapshots. This artifact affects neither the fractional flow rates as we see in the figure nor the relative L_1 norm of the error, which is shown to be around 0.07 for $\overline{S}$ and 0.01 for $\tilde{S}$ in table 4. We defer a more quantitative discussion of the error until the next section. The artifact is insignificant because it is in a region with slow flow. By resolving the fast channel we have sacrificed some resolution in the slow channels. Our coarse cell at the edge of the fast channel can be large and might contain the neighboring slow channel, which would explain such artifacts. What is more important though is that the overall accuracy is not compromised. Such artifacts will disappear and the fractional flow rates will be more accurate as we make the coarse block finer as in the series of saturation plots of figure 4.6.8. The corresponding fractional flow rates are in figure 4.6.9. We note that it is slightly harder to compute accurately the tail of the fractional flow curve than its part near the breakthrough.

FIGURE 4.6.2. Saturation snapshots for a layered permeability field, linear flux



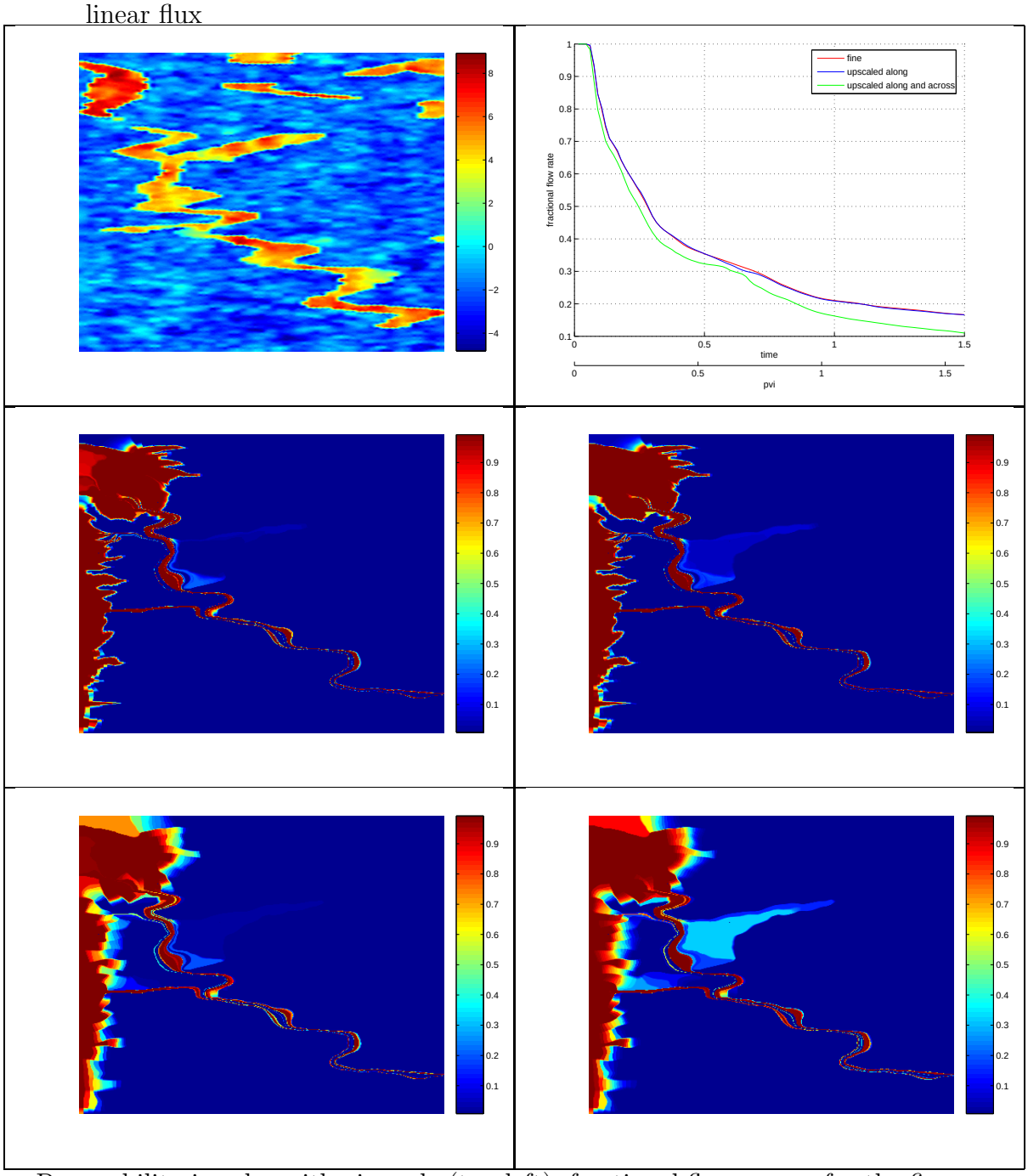
Permeability in a logarithmic scale (top left), fractional flow curves for the fine solution on a 400×400 grid and upscaled solutions on a 50×50 grid (top right), $\tilde{S}$ and $\overline{S}$ computed on a 50×50 grid with a moving mesh (middle and bottom right), and fine S averaged on the corresponding coarse blocks (middle and bottom left).

FIGURE 4.6.3. Saturation snapshots for the percolation case (Stanford 44), linear flux



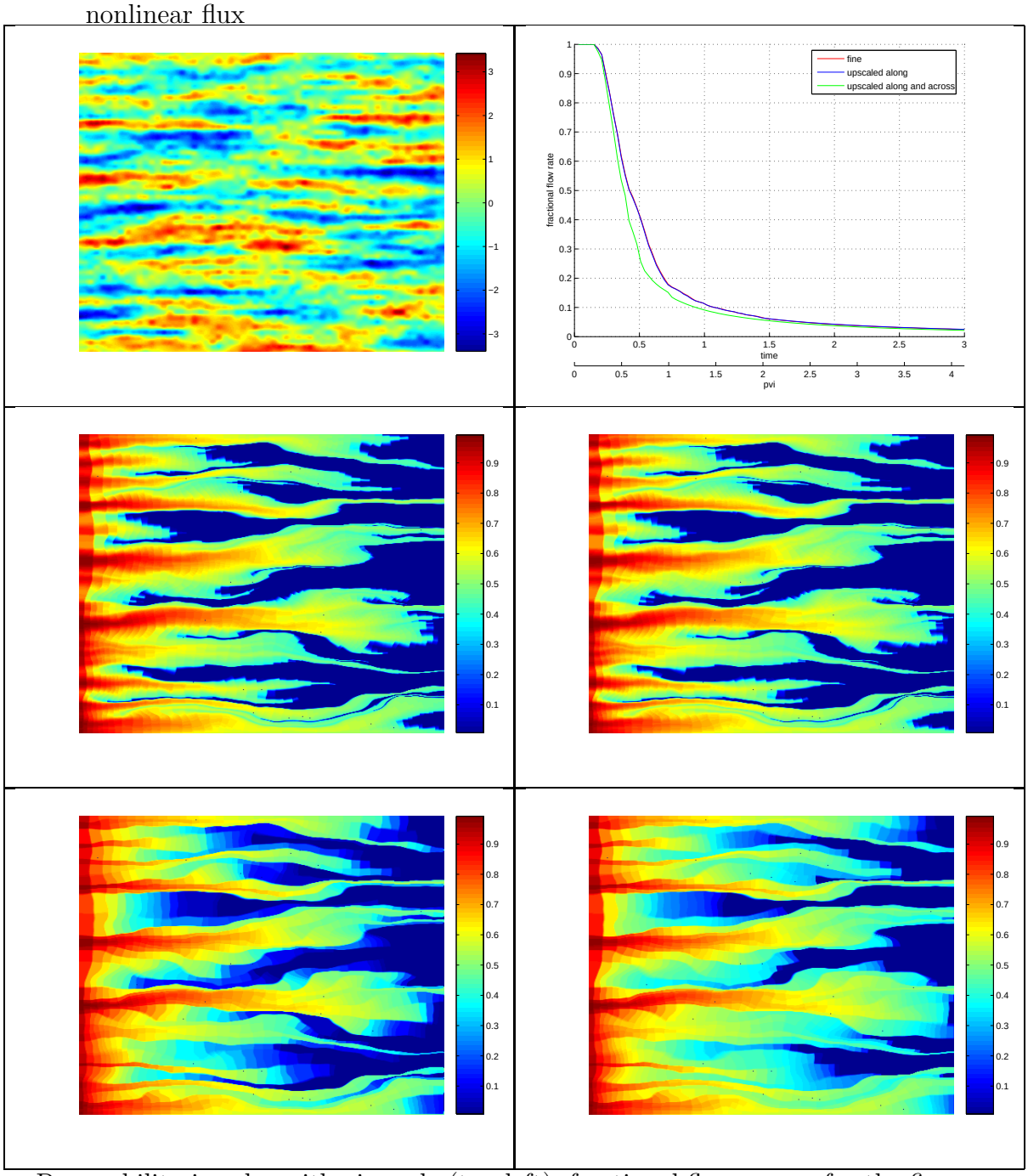
Permeability in a logarithmic scale (top left), fractional flow curves for the fine solution on a 400×400 grid and upscaled solutions on a 50×50 grid (top right), $\tilde{S}$ and $\overline{S}$ computed on a 50×50 grid with a moving mesh (middle and bottom right), and fine S averaged on the corresponding coarse blocks (middle and bottom left).

FIGURE 4.6.4. Saturation snapshots for a fast channel (SPE10 36), linear flux



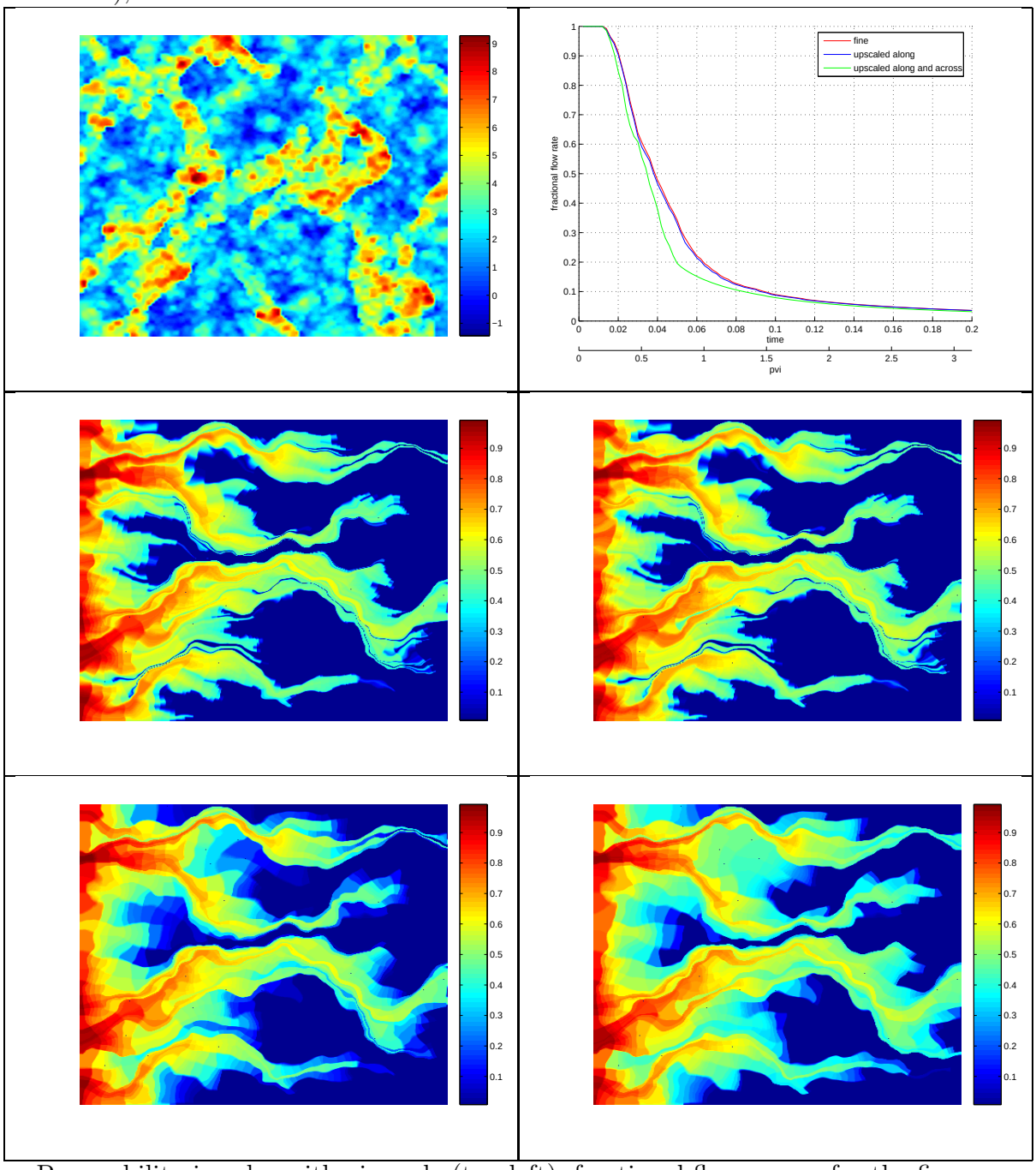
Permeability in a logarithmic scale (top left), fractional flow curves for the fine solution on a 400×400 grid and upscaled solutions on a 50×50 grid (top right), $\tilde{S}$ and $\overline{S}$ computed on a 50×50 grid with a moving mesh (middle and bottom right). and fine S averaged on the corresponding coarse blocks (middle and bottom left).

FIGURE 4.6.5. Saturation snapshots for a layered permeability field, nonlinear flux



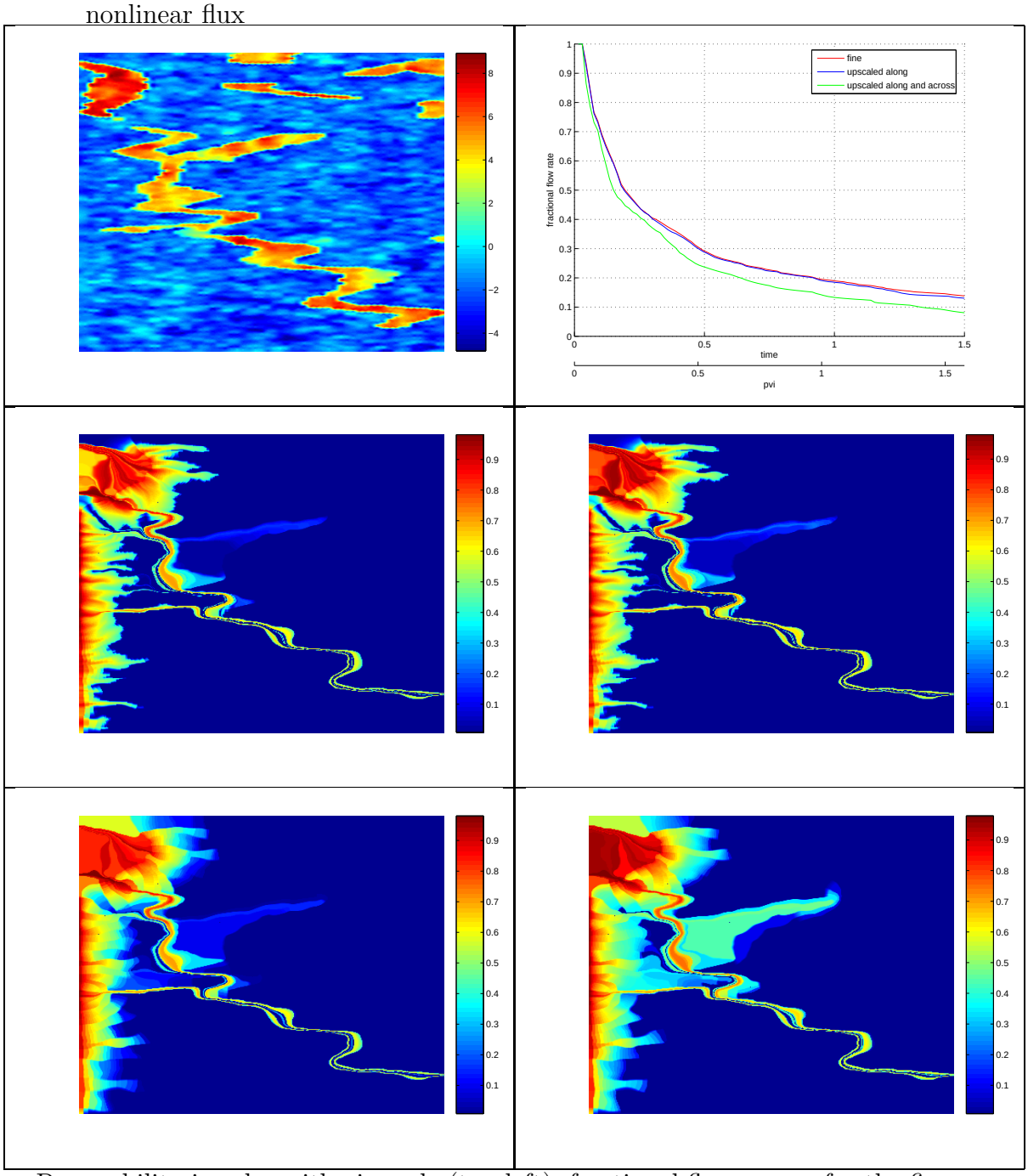
Permeability in a logarithmic scale (top left), fractional flow curves for the fine solution on a 400×400 grid and upscaled solutions on a 50×50 grid (top right), $\tilde{S}$ and $\overline{S}$ computed on a 50×50 grid with a moving mesh (middle and bottom right), and fine S averaged on the corresponding coarse blocks (middle and bottom left).

FIGURE 4.6.6. Saturation snapshots for the percolation case (Stanford 44), nonlinear flux



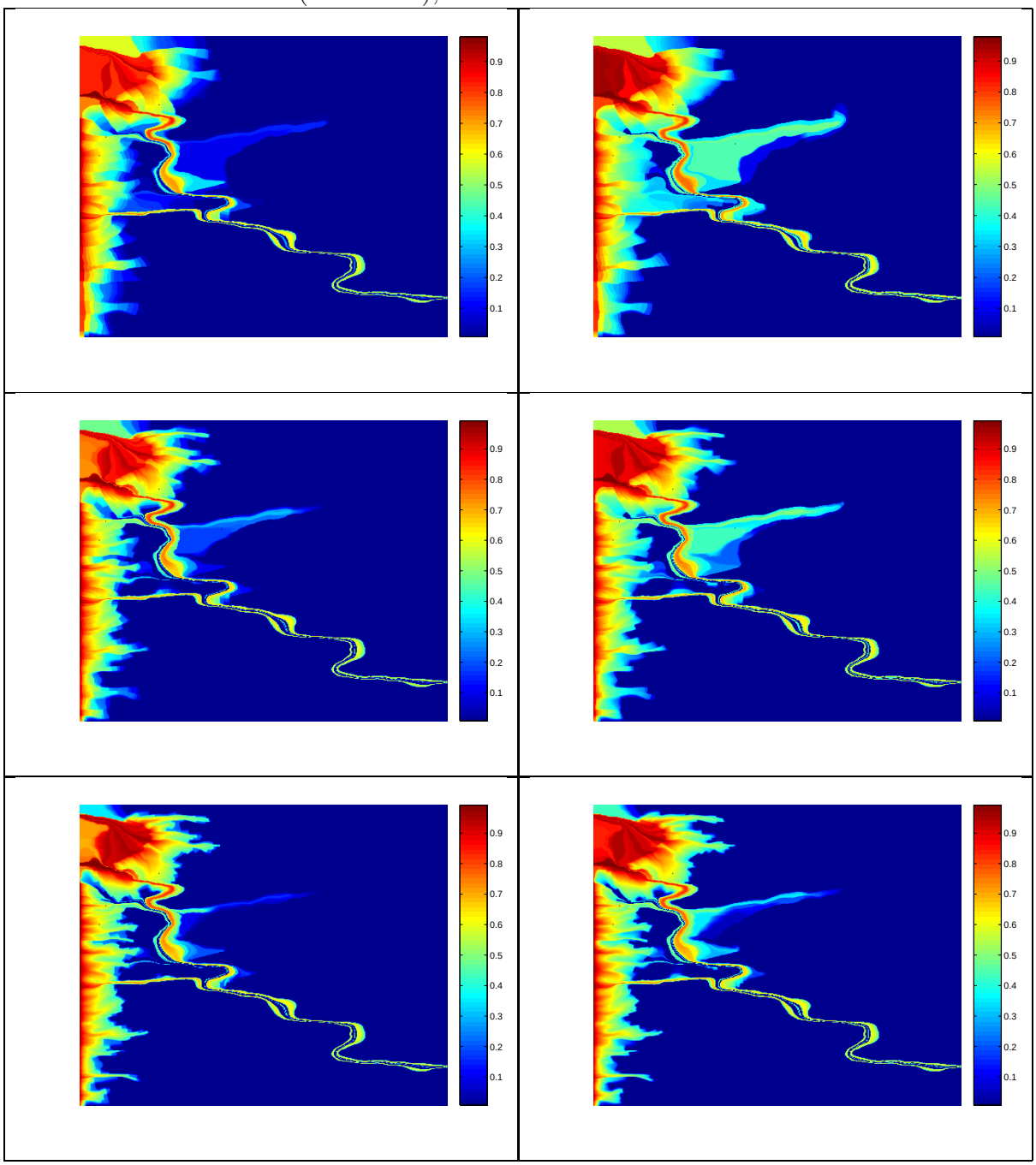
Permeability in a logarithmic scale (top left), fractional flow curves for the fine solution on a 400×400 grid and upscaled solutions on a 50×50 grid (top right), $\tilde{S}$ and $\overline{S}$ computed on a 50×50 grid with a moving mesh (middle and bottom right), and fine S averaged on the corresponding coarse blocks (middle and bottom left).

FIGURE 4.6.7. Saturation snapshots for a fast channel (SPE10 36), nonlinear flux



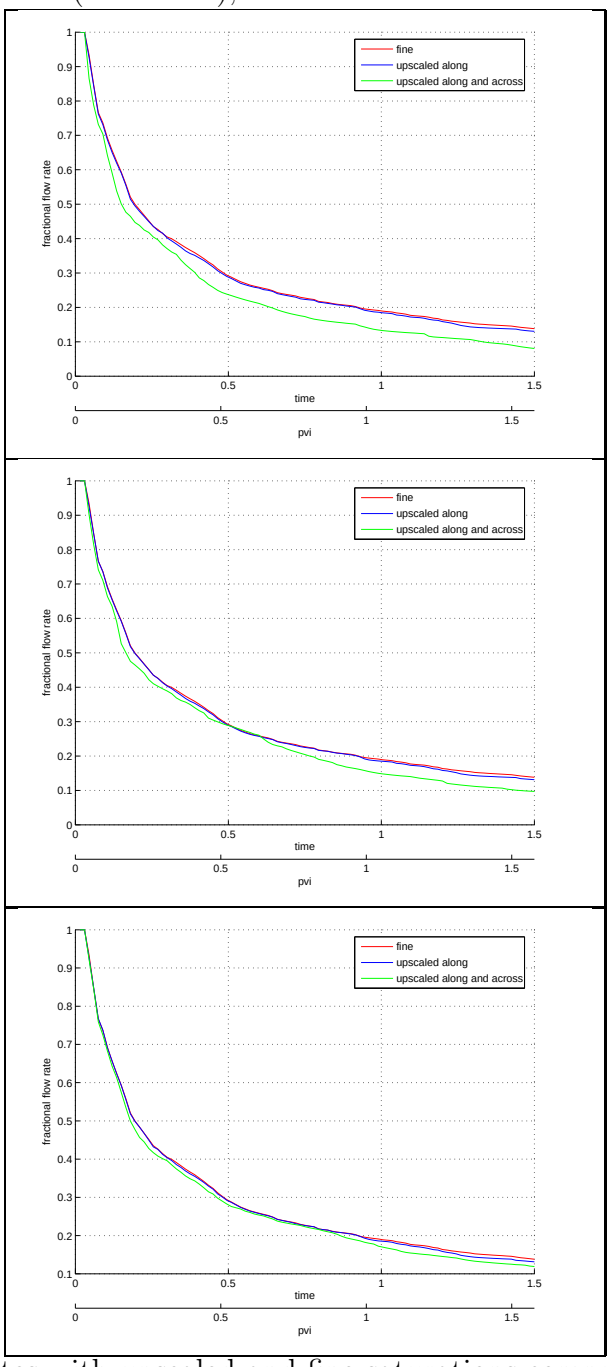
Permeability in a logarithmic scale (top left), fractional flow curves for the fine solution on a 400×400 grid and upscaled solutions on a 50×50 grid (top right), $\tilde{S}$ and $\overline{S}$ computed on a 50×50 grid with a moving mesh (middle and bottom right), and fine S averaged on the corresponding coarse blocks (middle and bottom left).

FIGURE 4.6.8. Saturation snapshots with decreasing coarse block size for a fast channel (SPE10 36), nonlinear flux



$\overline{S}$ computed on a 50×50, 100×100, 200×200 grid with a moving mesh (top, middle, and bottom right) and fine S computed on a 400×400 grid and averaged on the corresponding coarse blocks (top, middle, and bottom left), for same permeability as in figure 4.6.7.

FIGURE 4.6.9. Fractional flow rates with decreasing coarse block size for a fast channel (SPE10 36), nonlinear flux



Fractional flow rates with upscaled and fine saturations computed on a 50×50, 100×100, 200×200 grid (top, middle, and bottom right) for same permeability as in figure 4.6.7.

TABLE 2. Upscaling error for the layered permeability

LINEAR FLUX	25x25	50x50	100x100	200x200
L_1 error of $\tilde{S}$	0.0021	6.57×10^{-4}	2.15×10^{-4}	8.75×10^{-5}
L_1 error of $\overline{S}$ with macrodispersion	0.115	0.0696	0.0364	0.0135
L_1 error of $\overline{S}$ fine without macrodispersion	0.1843	0.0997	0.0505	0.0191

NONLINEAR FLUX	25x25	50x50	100x100	200x200
L_1 error of $\tilde{S}$	0.0023	8.05×10^{-4}	2.89×10^{-4}	1.29×10^{-4}
L_1 error of $\overline{S}$ with macrodispersion	0.116	0.0665	0.0433	0.0177
L_1 error of $\overline{S}$ fine without macrodispersion	0.151	0.0805	0.0432	0.0186

TABLE 3. Upscaling error for the percolation case (Stanford 44)

LINEAR FLUX	25x25	50x50	100x100	200x200
L_1 error of $\tilde{S}$	0.0222	0.0171	0.0122	0.0053
L_1 error of $\overline{S}$ with macrodispersion	0.0819	0.0534	0.0333	0.0178
L_1 error of $\overline{S}$ fine without macrodispersion	0.123	0.0834	0.0486	0.0209

NONLINEAR FLUX	25x25	50x50	100x100	200x200
L_1 error of $\tilde{S}$	0.0147	0.0105	0.0075	0.0040
L_1 error of $\overline{S}$ with macrodispersion	0.0842	0.0658	0.0371	0.0207
L_1 error of $\overline{S}$ fine without macrodispersion	0.119	0.0744	0.0424	0.0214

TABLE 4. Upscaling error for the fast channel (SPE10 36)

LINEAR FLUX	25x25	50x50	100x100	200x200
L_1 error of $\tilde{S}$	0.0128	0.0093	0.0072	0.0042
L_1 error of $\overline{S}$ with macrodispersion	0.0554	0.0435	0.0307	0.0176
L_1 error of $\overline{S}$ fine without macrodispersion	0.123	0.0798	0.0484	0.0258

NONLINEAR FLUX	25x25	50x50	100x100	200x200
L_1 error of $\tilde{S}$	0.0089	0.0064	0.0054	0.0033
L_1 error of $\overline{S}$ with macrodispersion	0.0743	0.0538	0.0348	0.0189
L_1 error of $\overline{S}$ fine without macrodispersion	0.0924	0.0602	0.0395	0.0202

4.6.3. Accuracy and Computational Cost. To compute the upscaling error we compare the upscaled solution computed on a 400×400 grid with the fine saturation computed on the 400×400 grid and averaged over the coarse grid. The errors are computed in the p, ψ frame, which is equivalent to weighting the error in every region with its flux. This choice is more meaningful physically. We display the upscaling error for the computations of the previous section in tables 2, 3, 4.

TABLE 5. Total error for the layered permeability

LINEAR FLUX	25x25	50x50	100x100	200x200
L_1 upscaling error of $\tilde{S}$	0.0021	6.57×10^{-4}	2.15×10^{-4}	8.75×10^{-5}
L_1 error of $\tilde{S}$ computed on coarse grid	0.0185	0.0062	0.0019	0.0015
L_1 upscaling error of $\overline{S}$	0.115	0.0696	0.0364	0.0135
L_1 error of computed on coarse grid	0.139	0.0779	0.0390	0.0144

NONLINEAR FLUX	25x25	50x50	100x100	200x200
L_1 upscaling error of $\tilde{S}$	0.0023	8.05×10^{-4}	2.89×10^{-4}	1.29×10^{-4}
L_1 error of $\tilde{S}$ computed on coarse grid	0.0268	0.0099	0.0027	9.38×10^{-4}
L_1 upscaling error of $\overline{S}$	0.116	0.0665	0.0433	0.0177
L_1 error of $\overline{S}$ computed on coarse grid	0.146	0.0797	0.0461	0.0184

TABLE 6. Total error for the percolation case (Stanford 44)

LINEAR FLUX	25x25	50x50	100x100	200x200
L_1 upscaling error of $\tilde{S}$	0.0222	0.0171	0.0122	0.0053
L_1 error of $\tilde{S}$ computed on coarse grid	0.0326	0.0161	0.0107	0.0113
L_1 upscaling error of $\overline{S}$	0.0819	0.0534	0.0333	0.0178
L_1 error of $\overline{S}$ computed on coarse grid	0.135	0.0849	0.0477	0.0274

NONLINEAR FLUX	25x25	50x50	100x100	200x200
L_1 upscaling error of $\tilde{S}$	0.0147	0.0105	0.0075	0.0040
L_1 error of $\tilde{S}$ computed on coarse grid	0.0494	0.0295	0.0150	0.0130
L_1 upscaling error of $\overline{S}$	0.0842	0.0658	0.0371	0.0207
L_1 error of $\overline{S}$ computed on coarse grid	0.17	0.11	0.0541	0.0303

To put these computations in perspective we note that a uniform computation with 400 points has a relative L_1 error of approximately 0.02 for the linear flux and 0.002 for the nonlinear flux according to earlier computations in tables 5, 6. $\tilde{S}$ is a very accurate approximation to the fine saturation. It is more accurate for the layered permeability than the other two fields because the variation of the velocity along the flow is smaller. $\overline{S}$ is less accurate because we are upscaling in two dimensions and not just one. The layered permeability has large variation across the flow so the effects of macrodispersion are more significant.

In tables 5, 6, 7 we show the total error, that is, the modeling and discretization error, in the case when we use a moving mesh to solve the saturation equation. It is interesting that the convergence of $\tilde{S}$ to S is observed even though the upscaling

TABLE 7. Total error for the fast channel (SPE10 36)

LINEAR FLUX	25x25	50x50	100x100	200x200
L_1 upscaling error of $\tilde{S}$	0.0128	0.0093	0.0072	0.0042
L_1 error of $\tilde{S}$ computed on coarse grid	0.023	0.0095	0.0069	0.0052
L_1 upscaling error of $\overline{S}$	0.0554	0.0435	0.0307	0.0176
L_1 error of $\overline{S}$ computed on coarse grid	0.0683	0.052	0.0361	0.0205

NONLINEAR FLUX	25x25	50x50	100x100	200x200
L_1 upscaling error of $\tilde{S}$	0.0089	0.0064	0.0054	0.0033
L_1 error of $\tilde{S}$ computed on coarse grid	0.0338	0.0148	0.0074	0.0037
L_1 upscaling error of $\overline{S}$	0.0743	0.0538	0.0348	0.0189
L_1 error of $\overline{S}$ computed on coarse grid	0.115	0.0720	0.0406	0.0204

TABLE 8. Computational cost

	fine $x.y$	fine p, ψ	$\tilde{S}$	$\overline{S}$
layered, linear flux	5648	257	9	1
layered, nonlinear flux	14543	945	28	4
percolation, linear flux	8812	552	12	1
percolation, nonlinear flux	23466	579	12	1
SPE10 36, linear flux	40586	1835	34	2
SPE10 36, nonlinear flux	118364	7644	25	2

It took 26 units of time to interpolate one quantity from the Cartesian to the pressure-streamline frame.

error is larger than the numerical error of the fine solution, which is 0.02 for the linear flux and 0.002 for the nonlinear flux in the L_1 norm, as mentioned before. The reason is that the location of the moving mesh points was selected so that the points are as dense near the shock as the fine solution using the parameter h_{min} (see section 3.3.4). This was done to observe the upscaling error clearly and also to have similar CFL constraints on the time step, which allows a clean comparison of computational times. We show the times required to compute the fractional flow curves above in table 8. The computations were performed on an AMD Athlon 1.5GHz with 512MB RAM. The upscaled solutions were computed on a 25×25 grid and the fine solution was computed on a 400×400 grid so we expect the $\overline{S}$ computations to be 256 times or more faster. The extra gain comes from a less restrictive CFL condition since we use an averaged velocity. The computations in the Cartesian frame are much slower; we believe that it is a combination of the fact that they have fluxes in two

TABLE 9. Comparison to a naive upscaling method for SPE10 36

	50x50	100x100	200x200
L_1error of $\overline{S}$	0.0720	0.0406	0.0204
cost of $\overline{S}$	4	36	253
L_1error of S	0.582	0.17	0.0955
cost of S	4	16	96

S is calculated on the coarse grid, by interpolating the fine permeability with a fast channel (SPE10 36) from 400×400 to the grid shown in each column.

dimensions and that our implementation was slower than the implementation in the pressure-streamline frame.

Finally we compare our method with a naive upscaling method. The naive upscaling method is simply interpolating the permeability field linearly onto the coarse grid and then solving for the saturation on that coarse grid. We use the same parameters as in the previous section and report the results for the permeability with a fast channel (SPE10 36) in table 9. To achieve an L_1 error of 0.09 in the final saturation our upscaling method requires a 50×50 grid and 4 units of time. If we tried to refine the grid we would require 200×200 points and 96 units of time. Our upscaling algoritm is about twenty times faster. We note that in table 9 as we double the number of grid points in each direction the total cost should increase by a factor of 8. We observe a slightly lower factor partly because the algorithm has an adaptive time step. For example, for the naive upscaling method from a 100×100 to a 200×200 grid the total cost increases by a factor of 6 instead of 8. The cost per time step for the 100×100 computation is 0.23 and the cost per time step for the 200×200 computation is 0.79. The cost per time step increases by 3.4 which is closer to the theoretical value of 4. The remaining discrepancy can be expained by the presence of some overhead cost in our implementation. The gain will be much larger if we use a Cartesian frame for the naive upscaling method, as is clear by the results of part 1 of this thesis and also of table 8. These numerical results can also be interpreted as a demonstration that the coarse grids don't resolve the small scales and that upscaling is necessary.

Chapter 5

Upscaling Two-Phase Flows

5.1. Pressure Equation

5.1.1. The Multiscale Finite Element Method (MSFEM). In the previous chapter we described a procedure to upscale the saturation equation. We approximated the fine saturation with its two-scale limit, which can be computed on a coarse grid. For the pressure equation it is not enough to upscale; we also want to be able to downscale, that is, given coarse quantities to compute fine quantities such as the fine pressure and velocity fields. The fine scale velocity is required for the computation of the effective velocity; it is an input of the upscaling scheme for the saturation. Hornung's book [**27**] is a standard reference for the foundation for upscaling elliptic equations.

Many methods have been proposed to upscale the pressure equation. They can be divided into two broad categories: "analytical upscaling methods" that assume a form for the upscaled pressure equation and derive its effective coefficients via ad hoc approximations and "numerical upscaling methods" where the upscaled equations are formed and solved numerically. For an extensive review we refer the reader to [**25**]. Here we mention an approach with elements related to our approach. In [**52**] Wen, Durlofsky, and Edwards propose a flow-based grid comprised of the streamfunction variable across the flow and a composite variable along the flow that is obtained by a weighted average of the pressure and the arclength variable and then apply a grid smoothing step to reduce its distortion. However they used this grid with an

analytical upscaling method. We will design a numerical upscaling method based on the MSFEM, which was introduced by Hou and Wu [**28**].

The MSFEM is particularly suitable for our purpose because it allows for downscaling. The idea is to define basis functions over coarse cells that satisfy the elliptic operator in a local problem subject to linear boundary conditions on the boundary of the cell. The basis functions resolve the fine scale. The operator that downscales all quantities from the coarse to the fine grid is given by the basis functions. Besides the theoretical importance of being able to compress multiscale operators using a multiscale basis set, the MSFEM can lead to an efficient upscaling method if we don't recompute the basis functions at every pressure time step. Then we have a method with a computational time that is a function of the number of coarse cells. The fact that the basis functions don't need to be updated has not been proven rigorously but has been observed in many two-phase flow experiments [**22**]. This experimental validation shows that the basis functions truly capture most of the fine scale behavior and that the MSFEM method rests on a sound idea.

As in every finite element method the continuous problem is discretized by projecting the solution onto a finite dimensional space that is spanned by the basis functions. The basis functions satisfy the same equation as the pressure on the coarse blocks of the domain. The main difficulty is to determine the boundary condition for the basis functions. A second drawback of the conforming MSFEM is the presence of resonance error in the basis functions near the cell boundary. This was noted by Hou, Wu, and Cai in [**28**], and they proposed the method of oversampling to improve on it [**23**]. However MSFEM with oversampling is non-conforming, which implies that the pressure is be continuous across the coarse blocks, and we cannot use it to compute the fine velocity. The mixed MSFEM of Chen and Hou [**9**] gives a conservative fine velocity within each basis and a coarse velocity that is also conservative. Their computations have excellent conservation properties. In our case we cannot use their algorithm because the fine velocity field will be nonconservative across coarse blocks, which can make the transformation singular. In this chapter we will see how to deal with both of these difficulties by changing to an adaptive coordinate frame.

5.1.2. The Modified MSFEM. It is well known that the domain of dependence for an elliptic operator is the whole domain over which the problem is defined. In practical terms this means that changing the value of the permeability at one point will affect the solution at every other point in the domain. In MSFEM methods we decompose the computation into a local part of computing the basis functions and a global part where all the local solutions are coupled together. The MSFEM basis functions are determined locally in each coarse cell and contain only information about the local structure of the operator to be upscaled.

For the case when the permeability has structures with long-range correlation that need to be captured, such as high permeability channels, Efendiev, Ginting, and Hou [**22**] proposed to incorporate more global information into the basis functions. They use the solution to the fine pressure problem at the initial time for supplying the boundary condition of the cell problem. Their method is referred to as the modified MSFEM. The motivation is that the connectivity of the fast channels will be reflected in the initial pressure and, through the cell boundary condition, in the basis functions as well. Even though the computation of the basis functions are decoupled in the sense that they can be done independent of each other, the basis functions contain some information about the global structure of the permeability.

The modified MSFEM is a conforming method and gives a continuous fine velocity field. Like the computation of the basis functions, computing the fine pressure initially is also a one time overhead of the method. In their study Efendiev, Ginting, and Hou observed that the modified MSFEM performs better than MSFEM. They were also able to demonstrate this analytically in certain cases. More importantly for us, the modified MSFEM performs better than MSFEM with oversampling, which means that it also removes the resonance error.

5.1.3. The Modified MSFEM in the Pressure-Streamline Frame. The framework of the modified MSFEM fits our purposes well. We want to tackle problems with fast channels, and we need to be able to compute the fine velocity field to upscale the saturation equation. We adapt it slightly at no extra computational cost to fit the

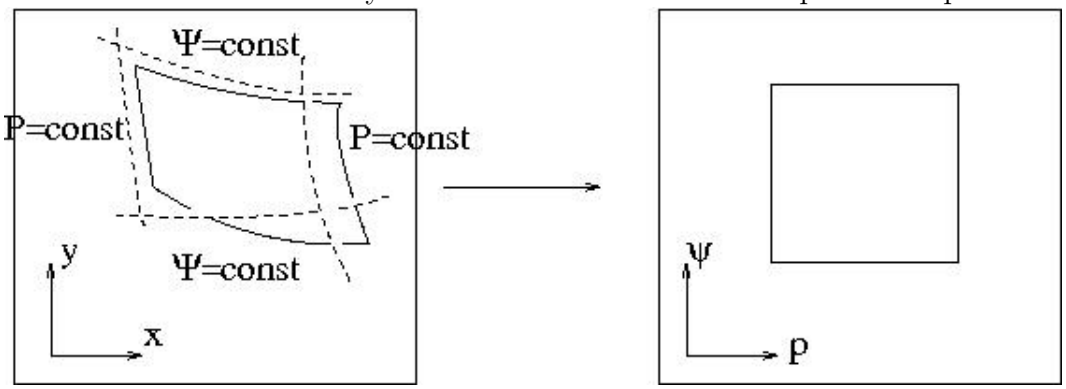
philosophy of our upscaling method for the saturation equation that was described in section 4.1.1.

For the saturation equation we picked the coarse blocks over which we average to have constant size in the pressure-streamline frame. We will follow the same idea for the pressure as well. Upscaling means finding the effective behavior or simply averaging, and our choice of frame will group regions of high and low velocity in separate cells. The variations in the velocity are to a large extent caused by the variations in λK. So we expect the variation of λK to be smaller in each coarse cell than if we had selected square cells in a Cartesian frame, which will lead to a smaller upscaling error for the pressure.

As in the modified MSFEM we solve for the initial pressure for the boundary condition for the bases. However we will then transform to the frame of the initial pressure P_0 and streamfunction Ψ_0 to pick the shape of the coarse blocks. The coarse blocks will be defined by the lines $p_0 = ih^c$, $\psi_0 = jh^c$. For the basis functions of the MSFEM, we select a linear boundary condition on edges with $p_0 = ih^c$ and no-flux boundary conditions on edges with $\psi_0 = jh^c$. Initially the exact solution of the pressure equation in the pressure-streamline frame is of course the identity $P(t = 0, p_0, \psi_0) = p_0$. This means the boundary condition for our basis functions coincides with the fine pressure solution initially, as in the modified MSFEM. In our method the long-range correlations of the basis functions are captured by the coordinate frame, whereas they are captured by the basis functions in the modified MSFEM.

If we assume that the pressure at time Δt has not changed much from the pressure at time 0, then the level sets $P(\Delta t, p = ih^c, \psi) = ih^c$ have not changed much either, and our boundary condition for the basis functions remains accurate. In figure 5.1.1 the solid lines show a coarse block in the (x, y) and $(p(0), \psi(0))$ frames. The dashed lines show the level sets of the pressure and streamfunction a short time afterwards. The are very close to the boundaries of the cell. A justification why this is so is that the pressure does not have a two-scale structure to lowest order for permeabilities with periodic fast variables according to homogenization theory. We cannot use the

FIGURE 5.1.1. Boundary conditions for the cells of the pressure equation



In a pressure streamline frame, lines of constant pressure and saturation form a uniform grid.

same line of reasoning for the no-flux boundary conditions because they depend on the derivative of the pressure.

5.1.4. Derivation of the Equations. We write the pressure equation in terms of the new variables. Let $\zeta = p_0$ and $\eta = \psi_0$. First we derive the elements of the Jacobian matrix and its inverse. They relate the differentials in x, y to differentials in ζ, η in the following way

$$\begin{pmatrix} dX \\ dY \end{pmatrix} = \begin{pmatrix} X_\zeta & X_\eta \\ Y_\zeta & Y_\eta \end{pmatrix} \begin{pmatrix} d\zeta \\ d\eta \end{pmatrix} = \mathbf{J}^{-1} \begin{pmatrix} d\zeta \\ d\eta \end{pmatrix}$$
$$\begin{pmatrix} d\zeta \\ d\eta \end{pmatrix} = \begin{pmatrix} \zeta_x & \zeta_y \\ \eta_x & \eta_y \end{pmatrix} \begin{pmatrix} dX \\ dY \end{pmatrix} = \mathbf{J} \begin{pmatrix} dX \\ dY \end{pmatrix}.$$

Solving the second equation for dX, dY we get

$$\begin{pmatrix} dX \\ dY \end{pmatrix} = \frac{1}{\zeta_x \eta_y - \zeta_y \eta_x} \begin{pmatrix} \eta_y & -\zeta_y \\ -\eta_x & \zeta_x \end{pmatrix} \begin{pmatrix} d\zeta \\ d\eta \end{pmatrix}.$$

Comparing the two equations for dX, dY we find the elements of the Jacobian matrix and its inverse

$$\begin{array}{cccccccccc} \zeta_x & = & \frac{u_0}{\lambda_0 K} & \zeta_y & = & \frac{v_0}{\lambda_0 K} & X_\zeta & = & \frac{u_0}{\|\mathbf{v_0}\|^2}\lambda_0 K & X_\eta & = & \frac{-v_0}{\|\mathbf{v_0}\|^2} \\ \eta_x & = & -v_0 & \eta_y & = & u_0 & Y_\zeta & = & \frac{v_0}{\|\mathbf{v_0}\|^2}\lambda_0 K & Y_\eta & = & \frac{u_0}{\|\mathbf{v_0}\|^2} \end{array}$$
$$J_0 = \frac{\|\mathbf{v_0}\|^2}{\lambda_0 K} \qquad J_0^{-1} = \|\mathbf{v_0}\|^{-2}\lambda_0 K.$$

We have used $\mathbf{v_0} = \lambda_0 K \nabla P_0$, $\nabla \Psi_0 = \mathbf{v_0^\perp}$. Using the chain rule on the pressure we find

$$\begin{pmatrix} P_\zeta \\ P_\eta \end{pmatrix} = \begin{pmatrix} X_\zeta & Y_\zeta \\ X_\eta & Y_\eta \end{pmatrix} \begin{pmatrix} P_x \\ P_y \end{pmatrix}.$$

Solving this equation for P_x, P_y we derive the rule for the transformation of derivatives

$$\begin{aligned} P_x &= \frac{1}{J_0^{-1}} \left((Y_\eta P_\zeta) - (Y_\zeta P_\eta) \right) \\ P_y &= \frac{1}{J_0^{-1}} \left(-(X_\eta P_\zeta) + (X_\zeta P_\eta) \right). \end{aligned}$$

We can take the pressure derivatives of the innermost parentheses outside the parentheses

$$\begin{aligned} P_x &= \frac{1}{J_0^{-1}} \left((Y_\eta P)_\zeta - (Y_\zeta P)_\eta \right) \\ P_y &= \frac{1}{J_0^{-1}} \left(-(X_\eta P)_\zeta + (X_\zeta P)_\eta \right). \end{aligned}$$

Applying this rule twice we get

$$\begin{aligned} \partial_x \lambda K P_x &= \frac{1}{J_0^{-1}} \left((Y_\eta \lambda K \frac{1}{J_0^{-1}} \left((Y_\eta P)_\zeta - (Y_\zeta P)_\eta \right))_\zeta - (Y_\zeta \lambda K \frac{1}{J_0^{-1}} \left((Y_\eta P)_\zeta - (Y_\zeta P)_\eta \right))_\eta \right) \\ \partial_y \lambda K P_y &= \frac{1}{J_0^{-1}} \left(-(X_\eta \lambda K \frac{1}{J_0^{-1}} \left(-(X_\eta P)_\zeta + (X_\zeta P)_\eta \right))_\zeta + (X_\zeta \frac{1}{J_0^{-1}} \left(-(X_\eta P)_\zeta + (X_\zeta P)_\eta \right))_\eta \right). \end{aligned}$$

We can write the transformed operator using the metric tensor

$$\nabla \lambda K \nabla P \longmapsto \frac{1}{J_0^{-1}} \nabla_{\zeta,\eta} \begin{bmatrix} \lambda K \frac{Y_\eta^2 + X_\eta^2}{J^{-1}} & \lambda K \frac{-Y_\eta Y_\zeta - X_\eta X_\zeta}{J^{-1}} \\ \lambda K \frac{-Y_\eta Y_\zeta - X_\eta X_\zeta}{J^{-1}} & \lambda K \frac{Y_\zeta^2 + X_\zeta^2}{J^{-1}} \end{bmatrix} \nabla_{\zeta,\eta} P.$$

Using the expressions for the elements of the Jacobian matrix we find

$$\frac{1}{J_0^{-1}} \nabla_{\zeta,\eta} \begin{bmatrix} \frac{\lambda}{\lambda_0} & 0 \\ 0 & \lambda \lambda_0 K^2 \end{bmatrix} \nabla_{\zeta,\eta} P.$$

The saturation equation becomes

$$S_t + (\mathbf{v} \cdot \nabla P_0) S_\zeta + (\mathbf{v} \cdot \nabla \Psi_0) S_\eta = 0.$$

We propose a second transformation to solve the saturation in the frame of P,Ψ. It is defined in terms of the velocity $\mathbf{v}_{\zeta,\eta}$ in the ζ,η frame with $\mathbf{v}_{\zeta,\eta} = A_o \nabla_{\zeta,\eta} P$, $\nabla_{\zeta,\eta}\Psi_0 = \mathbf{v}^{\perp}_{\zeta,\eta}$. The chain rule gives a useful relation

$$\begin{aligned}
\mathbf{v}_{\zeta,\eta} &= A_o \nabla_{\zeta,\eta} P = A_0 \begin{bmatrix} X_\zeta & Y_\zeta \\ X_\eta & Y_\eta \end{bmatrix} \nabla P \\
&= \begin{bmatrix} \frac{\lambda}{\lambda_0} & 0 \\ 0 & \lambda\lambda_0 K^2 \end{bmatrix} \begin{bmatrix} \frac{u_0}{\|\mathbf{vo}\|^2}\lambda_0 K & \frac{v_0}{\|\mathbf{vo}\|^2}\lambda_0 K \\ \frac{-v_0}{\|\mathbf{vo}\|^2} & \frac{u_0}{\|\mathbf{vo}\|^2} \end{bmatrix} \nabla P \\
&= \lambda K J_0^{-1} \begin{bmatrix} \frac{u_0}{\lambda_0 K} & \frac{v_0}{\lambda_0 K} \\ -v_0 & u_0 \end{bmatrix} \nabla P = J_0^{-1} \begin{bmatrix} \mathbf{v}\cdot\nabla P_0 \\ \mathbf{v}\cdot\nabla\Psi_0 \end{bmatrix} \\
\Rightarrow \quad J_0 \mathbf{v}_{\zeta,\eta} &= \begin{bmatrix} \mathbf{v}\cdot\nabla P_0 \\ \mathbf{v}\cdot\nabla\Psi_0 \end{bmatrix}.
\end{aligned}$$

Using this relation we rewrite the velocity $(\mathbf{v}\cdot\nabla P_0, \mathbf{v}\cdot\nabla\Psi_0)\cdot\nabla_{\zeta,\eta}P$ in the P, Ψ frame, and we arrive at the following saturation equation

$$S_t + J_0 \mathbf{v}_{\zeta,\eta} A_o^{-1} \mathbf{v}_{\zeta,\eta} S_p = 0.$$

The full algorithm is shown in figure 5.1.2. There are three different time steps involved in the algorithm: Δt_B, the time between two updates of the bases, Δt_P, the time between two updates of the pressure, and the time step of the saturation equation. In practice we rarely have to update the basis functions. In all experiments that follow, updating the basis functions lead to no significant improvement.

5.2. Implementation

5.2.1. Computing the Pressure with MSFEM. As in the FV method we divide the domain Ω into coarse primal and dual cells. Both the coarse primal cells and the points at their centers are denoted by $1,\ldots,9$. Similarly the points at the center of the coarse dual cells are denoted by $A,\ldots,D$ and so are the centers of the coarse dual cells as in figure 5.2.1. For each coarse cell of the dual volume we compute four basis functions ϕ from which we construct the finite element basis χ. For example for cell A we solve

$$\nabla\lambda(S) K \nabla\phi_A^k = 0 \text{ in } A,$$

FIGURE 5.1.2. The Multiscale Streamline Method

(1) Solve for the initial pressure $P(0)$ and streamfunction $\Psi(0)$ on the fine grid. This is a one time overhead of the method that provides exact boundary conditions for the cell problems at $t = 0$.
(2) Transform the velocity and the saturation to the frame of $p(0), \psi(0)$.
(3) Advance the saturation equation

$$S_t + \frac{|\mathbf{v}|^2}{\lambda_0 K} S_\zeta = 0$$

until time T.
(4) Solve for the basis functions of the MSFVM method.
(5) Advance the equation until $t = \Delta t_B$, with the following steps:
   (a) Solve for the new pressure $P(T)$ using the old basis functions

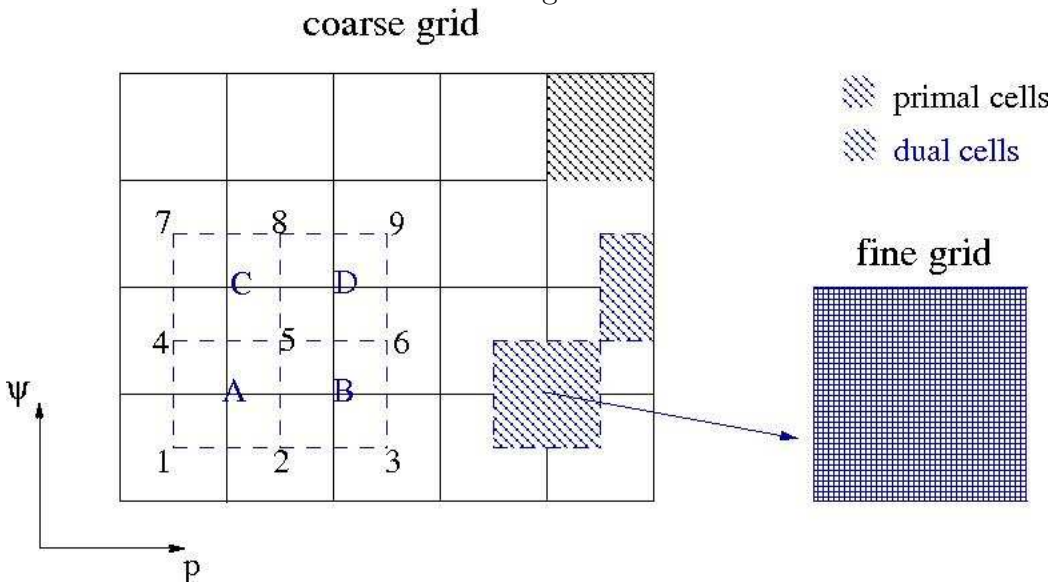
$$\frac{\partial}{\partial \zeta}\left(\frac{\lambda(S(T))}{\lambda(S(0))}\frac{\partial P(T)}{\partial \zeta}\right) + \frac{\partial}{\partial \eta}\left(\lambda(S(0))\lambda(S(T)K^2\frac{\partial P(T+\Delta t)}{\partial \eta}\right) = 0.$$

   (b) Transform the velocity and the saturation to the frame $P(T), \Psi(T)$.
   (c) Advance the saturation equation

$$S_t + J_0 A_o^{-1} \|\mathbf{v}_{\zeta,\eta}\|^2 S_p = 0$$

   until time $T + \Delta t_P$.
   (d) Transform the saturation back to the frame $P(0), \Psi(0)$.
(6) Go to step 4 until $t = T_F$.

FIGURE 5.2.1. Coarse and fine grids for the MSFVM Method



subject to $\phi_A^k = \delta_{ik}$ on the corners of A, linear boundary condition on the edge $14, 25$ along constant pressure and no flux on the edges $12, 45$ along the same streamline. The basis function χ^5 is defined on $A \bigcup B \bigcup C \bigcup D$ by ϕ_A^5, ϕ_B^5, ϕ_C^5, ϕ_D^5. Denote with H^1

the space of functions whose derivative is in L_2 that vanish on the $p = 0, 1$ boundary. When we look at the continuous pressure equation as a variational problem in H_1 its solution satisfies

$$\int \lambda K \nabla P \cdot \nabla \phi = 0, \forall \phi \in H_1.$$

We discretize this equation by considering it in the space $V = span\{\chi^k\}$. Then we can write P, ϕ as a linear combination of the basis functions with coefficients p_k, ϕ_k. The solution of the discrete variational problem satisfies

$$\sum_k p_k \int \lambda K \nabla \chi^k \cdot \nabla \chi^l = 0, \forall \chi^l \in V. \tag{5.2.1}$$

5.2.2. Computing the Streamfunction.

With the Multiscale Finite Volume Method (MSFVM). The solution of 5.2.1 will have a continuous velocity inside the dual volumes but not on their boundaries. As we noted in section 3.3.1, the numerical velocity must satisfy the discrete conservation (2.2.1) so that the streamfunction is uniquely defined and so that the image of the unit square in the pressure-streamline frame is a rectangle. In the algorithm that we presented in the previous chapter for one-phase flow this was achieved by a finite volume discretization of the pressure equation. For the upscaled scheme we will borrow ideas from the multiscale finite volume method (MSFVM).

To compute a continuous conservative velocity field, Jenny, Lee, and Tschelepi [**33**] used a second set of basis functions $\overline{\phi}$ defined on the primal grid. For example $\overline{\phi}^5$ is defined on cell 5 and satisfies

$$\nabla \lambda(S) K \nabla \overline{\phi}^5 = 0 \text{ in } 5.$$

The fluxes on the boundary are provided by the fluxes of χ^5. Then a conservative fine scale velocity field can be extracted from the pressure. Like the basis functions of the MSFVM method this second set of basis functions needs to be computed in a preprocessing step only.

With an Elliptic Equation. A second way to obtain the streamfunction is the following elliptic equation for Ψ

$$\nabla^2 \Psi = -\nabla^{\perp} \cdot \mathbf{v}.$$

This can be done very efficiently by substituting $\Psi = \breve{\Psi} + y$. Then the boundary condition for $\breve{\Psi}$ is homogeneous and the resulting equation can be solved very efficiently with the Fast Fourier Transform in $O(Nlog(N))$ operations, where N is the total number of fine grid points.

Obtaining the streamfunction Ψ with an elliptic equation does not require that the velocity be incompressible on a discrete level. However it can introduce a larger numerical error than obtaining the streamfunction by integrating the velocity of the previous section. The reason is that the right hand side of the elliptic equation involves derivatives of the velocity, whereas the approach with the dual bases involves only the velocity. In general the velocity will not be smooth and the computation of the curl will be inaccurate. This effect is alleviated by the fact that the transformation in the initial pressure and streamfunction coordinates is near the identity. Of course we have no theoretical proof for this claim.

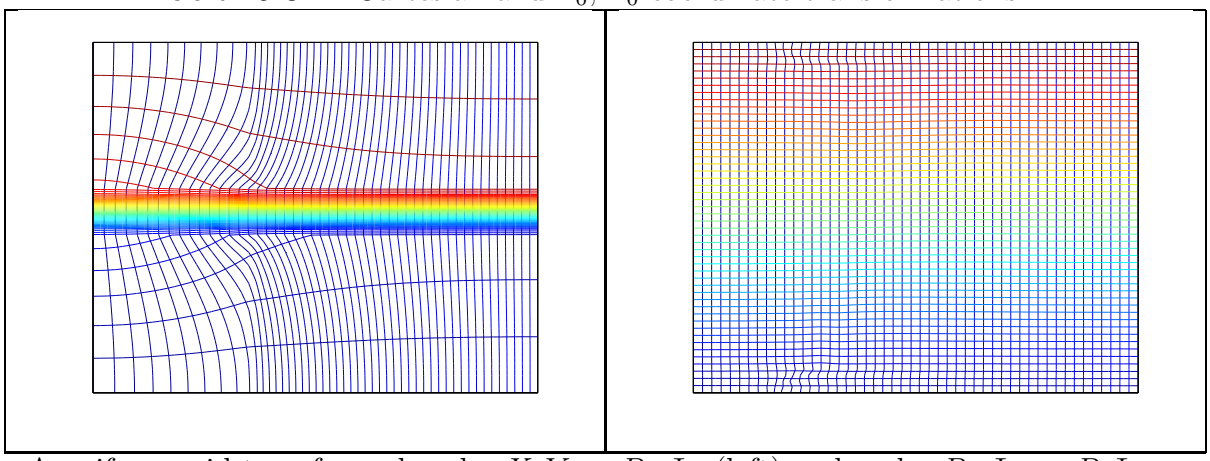
5.2.3. Coarse Interpolation. We have presented all the ingredients for an efficient scheme for the two-phase flow equations. Before we put them together in the next section, one final comment on coarse interpolation is appropriate. The only part of the algorithm that contains a computation on the fine scale is the computation of $\overline{S}(p,\psi)$ and $\overline{\tilde{v}_0}(p,\psi)$, which are used to solve the saturation equation. Since these quantities are defined on the coarse grid, the question arises whether we really have to go on the fine scale to compute them. This is possible for $\overline{S}(p,\psi)$ by first writing the Jacobian of the transformation in terms of the multiscale basis functions χ^k of the pressure.

$$dPd\Psi = \frac{|\mathbf{v}|^2}{\lambda K} dXdY = \sum_{\text{fine nodes of cell}} P_k P_l \lambda K \nabla\chi^k \cdot \nabla\chi^l dXdY.$$

Then

$$\overline{S}(p,\psi) \;=\; \int_{\text{cell}} S(p,\psi) dPd\Psi \;=\; \sum_{\text{fine nodes of cell}} P_k P_l \int \lambda K \nabla\chi^k \cdot \nabla\chi^l dXdY.$$

FIGURE 5.3.1. Cartesian and P_0, Ψ_0 coordinate transformations



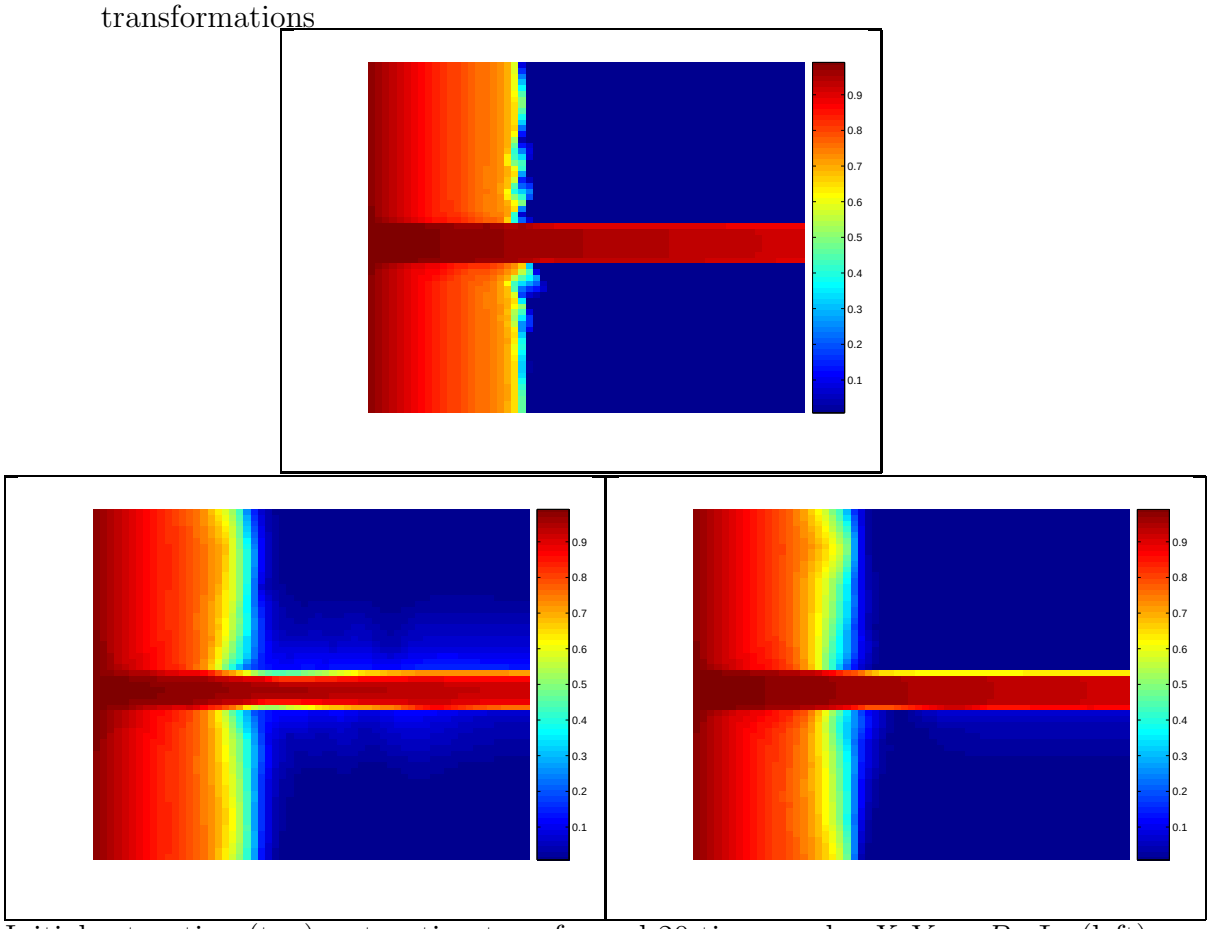
A uniform grid transformed under $X, Y \to P_0, \Psi_0$ (left) and under $P_0, \Psi_0 \to P, \Psi$ (right)

The velocity is not a simple arithmetic average but an arithmetic average across the flow lines and a harmonic average along the flow lines and cannot be expressed in terms of the multiscale basis functions.

5.3. Numerical Results

5.3.1. The Transformation $P_0, \Psi_0 \to P, \Psi$. We want to demonstrate that the map $P_0, \Psi_0 \to P, \Psi$ is very close to the identity and that this leads to less numerical diffusion. We consider a permeability field with a straight fast channel, nonlinear flux with $m = 1$. We compute the saturation profile until $T = 0.3$ and denote the pressure and streamfunction at that time by P_0, Ψ_0. We then advance the saturation until $T + \Delta t_P$ with $\Delta t_P = 0.0001$ and denote the pressure and streamfunction at that time by P, Ψ. In figure 5.3.1 it is obvious that $P_0, \Psi_0 \to P, \Psi$ is very close to the identity. We transform the saturation 20 times back and forth and show the results in figure 5.3.2 . There is more diffusion both in the fast channel and the uniform front when the transformation is far from the identity. The relative L_1 error of the saturation transformed under $X, Y \to P_0, \Psi_0$ was 0.0971 and under $P_0, \Psi_0 \to P, \Psi$ it was 0.0584. We did not observe such a clear difference in more realistic flow profiles but we believe this is due to the fact that our method to transform the saturation was not optimized at all to preserve sharp shocks.

FIGURE 5.3.2. Numerical diffusion in Cartesian and P_0, Ψ_0 coordinate transformations



Initial saturation (top), saturation transformed 20 times under $X, Y \to P_0, \Psi_0$ (left), and under $P_0, \Psi_0 \to P, \Psi$ (right).

5.3.2. Upscaling Only the Pressure. We are interested in isolating the pressure upscaling error, so we perform computations with the fine saturation. We will use the layered permeability field, the saturation equation with nonlinear flux, and $m = 1$. All norms are in the p_0, ψ_0 variables because the quantities that enter our numerical method depend on p_0, ψ_0 and not x, y. A comparison at $t = 0$ is meaningless because initially both the upscaled and fine methods use the fine pressure to compute the velocity, that is, initially the two methods coincide. We have recomputed the pressure four times and compare the upscaled and fine quantities at one pressure time step before the final time. This ensures that $\lambda(S)K$ at $t = \frac{3T_{final}}{4}$ is much different than at time $t = 0$, which is what we used to compute the basis functions. Then we have a strict test whether our upscaling method is accurate and whether the basis

TABLE 1. Pressure upscaling error for the layered permeability

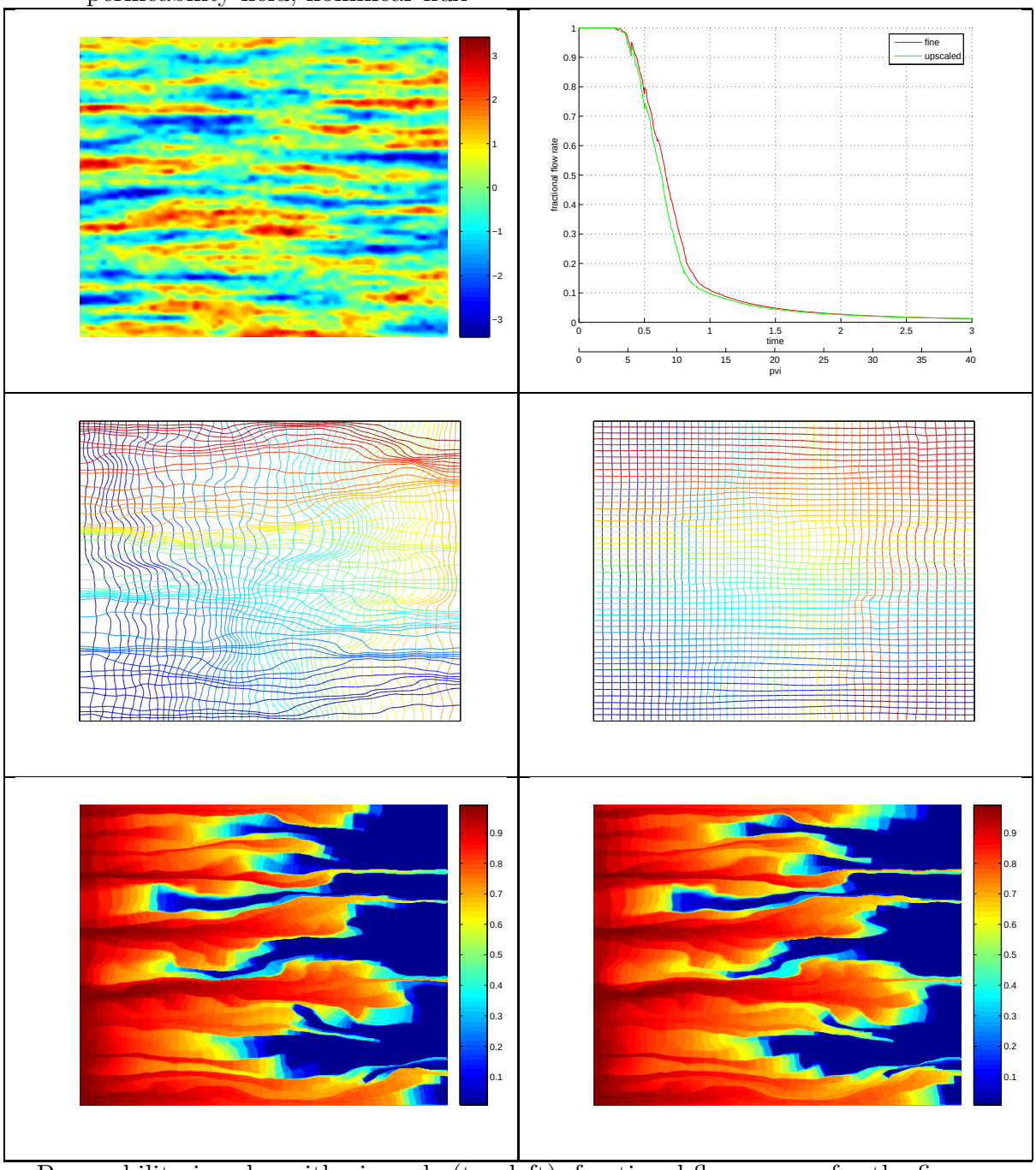
	50x50	100x100	200x200
L_2 pressure error at $t = \frac{3T_{final}}{4}$	0.0013	0.0007	0.0004
L_2 velocity error at $t = \frac{3T_{final}}{4}$	0.0250	0.0149	0.0076
L_1 saturation error $t = T_{final}$	0.0108	0.0056	0.0028

These errors should be compared with the change in velocity between $t = 0$ and $t = \frac{3T_{final}}{4}$. The reason is that the upscaled scheme computes fine quantities at $t = 0$. The norms are computed in the p_0, ψ_0 variables. The L_2 difference of the fine velocity at $t = 0$ and $t = \frac{3T_{final}}{4}$ was 0.2041.

functions need to be updated. To put the results in perspective in the caption we report how much the velocity changed from $t = 0$ to $t = \frac{3T_{final}}{4}$. In table 1 we show the results. The error in velocity is larger than in pressure because the pressure is smoother. Overall our upscaling error for the pressure is much smaller than for the saturation.

5.3.3. Full Upscaling. In the results that follow we use the layered permeability field, the saturation equation with nonlinear flux, and $m = 1$. The final time was $T_{final} = 0.4$ and the pressure time step was $\Delta t_P = 0.1$, so the pressure was computed four times during the simulation. We will use the full algorithm 5.1.2 which upscales both pressure and saturation with $\overline{S}$ or $\tilde{S}$. We show some plots in the case when the fine solution is defined on a 400×400 grid and the upscaled on a 50×50 grid using $\overline{S}$ in figure 5.3.3.

FIGURE 5.3.3. Snapshots for the full upscalling method for a layered permeability field, nonlinear flux



Permeability in a logarithmic scale (top left), fractional flow curves for the fine solution on a 400×400 grid and upscaled solutions on a 50×50 grid (top right), coordinate transformations $X, Y \rightarrow P, \Psi$(middle left) and $P_0, \Psi_0 \rightarrow P, \Psi$(middle right), $\overline{S}$ computed on a 50×50 grid (bottom right), and fine S averaged on the corresponding coarse blocks (bottom left).

TABLE 2. Convergence of the full upscaling method for the layered permeability

with $\tilde{S}$	50x50	100x100	200x200
L_2 pressure error at $t = \frac{3T_{final}}{4}$	0.0014	0.007	0.004
L_2 velocity error at $t = \frac{3T_{final}}{4}$	0.0235	0.0137	0.0072
L_1 saturation error $t = T_{final}$	0.0105	0.0052	0.0027

with $\overline{S}$	50x50	100x100	200x200
L_2 pressure error at $t = \frac{3T_{final}}{4}$	0.0046	0.0021	0.0008
L_2 velocity error at $t = \frac{3T_{final}}{4}$	0.0530	0.0335	0.0246
L_1 saturation error $t = T_{final}$	0.0546	0.0294	0.0134

The pressure and velocity errors should be compared to the change in velocity between $t = 0$ and $t = \frac{3T_{final}}{4}$. The reason is that the upscaled scheme computes fine quantities at $t = 0$. The norms are computed in the p_0, ψ_0 variables. The L_2 difference of the fine velocity at $t = 0$ and $t = \frac{3T_{final}}{4}$ was 0.2041.

TABLE 3. Comparison to a Cartesian upscaling method

	Cartesian upscaling	with $\overline{S}$	with $\tilde{S}$
L_1 saturation error $t = T_{final}$on 50x50	0.164	0.0546	0.0105

The fine solution was defined on a 400×400 grid and the upscaled on a 50×50 grid.

We study the convergence of the upscaled scheme to the fine solution when the fine solution is defined on a 400×400 grid in table 2. We also compare our upscaling scheme to a simple upscaling method using the coarse velocities of the MSFVM method in the cartesian frame.3. As in the one-phase flow we observe that upscaling with $\tilde{S}$ is more accurate than $\overline{S}$. In order not to be wasting computational resources, the pressure upscaling error should be similar in magnitude to the saturation upscaling error. If we upscale the pressure to 50×50 points then we should use $\tilde{S}$ instead of $\overline{S}$ because the saturation error with $\overline{S}$ quintuples whereas with $\tilde{S}$ it remains the same as is obvious from tables 1 and 2. Another way to control the upscaling error is of course through the size of the coarse cells. We could upscale for example the saturation with $\overline{S}$ to 50×50 cells and the pressure to 20×20 cells.

Chapter 6

Conclusion and Future Directions

The method that we have presented here can be made more accurate in a number of ways. It can be improved by an interpolation method that preserves shocks so that changing coordinates does not introduce too much diffusion to the saturation. An option is to track the shock separately and then use one-sided interpolation near the shock. We have observed that the coefficients of the elliptic equation in the p_0, ψ_0 frame are singular; perhaps there is a way to regularize them, or regularize the transformation without introducing too much error. In particular the square of the permeability appears so this regularization will only be a simple rescaling of the permeability tensor. We note that it is possible to write an elliptic equation for the pressure in the pressure-streamline frame, an equation directly for $X(p, \psi)$ which suffers from the same drawback. Its permeability depends on the solution and the coefficients are singular as well.

Strictly speaking it is not possible to extend this method to general three-dimensional flows because the Jacobian of transformation defined by the pressure and the two streamfunctions can be negative. This is not necessarily a difficult obstacle. An approximate transformation could be used in such cases. The questions that arise naturally are in which cases the approximation does not lead to a large error and what that error is.

What is different with our approach is that we are able to get rid of the awkward assumption of periodicity and scale separation. This is pleasing on a theoretical level but we believe that it can also explain why the numerical method is so accurate. It would be interesting to use a coordinate transformation to obtain the two-scale limit

under stochastic homogenization and try to relax some of its awkward assumptions. In this work we tackled the homogenization of the saturation equation assuming that the velocity does not depend on the saturation. Effectively we treated the coupling between the pressure and saturation equations by linearization. This is equivalent to considering one-phase flow. Further work should address the homogenization of the coupled two-phase flow equations and investigate whether any new physical phenomena arise. More generally it is interesting to apply the same philosophy to other equations, either in cases when a transformation groups the small scale in a more intuitive way or equations that are reduced to a simpler form under a coordinate transformation.

When I heard the learned astronomer,
When the proofs, the figures, were ranged in columns before me,
When I was shown the charts and diagrams, to add, divide, and measure them,
When I sitting heard the astronomer where he lectured with much applause in the lecture room,
How soon unaccountable I became tired and sick,
Till rising and gliding out I wandered off by myself,
In the mystical moist night-air, and from time to time,
Looked up in perfect silence at the stars.

Walt Whitman

Bibliography

[1] Arbogast TJ, The Existence of Weak Solutions for Single Porosity and Simple Dual-Porosity Models of Two-Phase Incompressible Flow, Nonlinear Analysis, Vol. 19, p. 1009-1031 (1992)

[2] Amaziane B, Bourgeat A, Koebbe J, Numerical Simulation and Homogenization of Two-Phase Flow in Heterogeneous Porous Media, Transport in Porous Media, Vol. 6, p. 519-547 (1991)

[3] Barenblatt GI, Patzek TW, Sillin DB, The Mathematical Model of Non-Equilibrium Effects in Water-Oil Displacement, SPE 75169 (2002)

[4] Batycky PR, A Three-Dimensional Two-Phase Field Scale Streamline Simulator, Stanford Thesis (1997)

[5] Bear J, Dynamics of Fluids in Porous Media, American Elsevier Publishing Company (1972)

[6] Blunt MJ, Lui K, Thiele MR, A Generalized Streamline Method to Predict Reservoir Flow, Vol. 2, p. 259-269 (1996)

[7] de Boor C, Good Approximation by Splines with Variable Knots II, Proceedings of Conference on the Numerical Solution of Differential Equations (University of Dundee, Dundee 1973), Lecture Notes in Mathematics 363, Springer-Verlag Berlin, p. 12-20 (1974)

[8] Buckley SE, Leverett MC, Mechanisms of Fluid Displacements in Sands, AIME Transactions, Vol. 147, p. 107-116 (1942)

[9] Chen C, Hou TY, A Mixed Finite Element Method for Elliptic Problems With Rapidly Oscillating Coefficients, Mathematics of Computation, Vol. 72, p. 541-576 (2003)

[10] Chen Y, Durlovsky LJ, Adaptive Local-Global Upscaling for General Flow Scenarios in Heterogeneous Formations, Transport in Porous Media, to appear

[11] Christie M, Blunt M, Tenth SPE Comparative Solution Project. A Comparison of Upscaling Techniques, SPE Reserv. Eval. Eng., No. 4, p. 308-317 (2001)

[12] Colella P, Multidimensional Upwind Methods for Hyperbolic Conservation Laws, Journal of Computational Physics, Vol. 87, p. 171-200 (1990)

[13] Dafermos Constantine M, Hyperbolic Conservation Laws in Continuum Physics, Springer-Verlag Berlin Heidelberg (2000)

[14] Dafermos C, Polygonal Approximations to Solutions of the Initial Value Problem for a Conservation Law, Journal of Mathematical Analysis and Applications, Vol. 38, p. 33-41 (1973)

[15] Datta-Gupta A, King MJ, A Semianalytic Approach to Tracer Flow in Heterogeneous Permeable Media, Advances in Water Resources, 18, 9-24 (1995)

[16] Deutsch C, Journel AG, GSLIB: Geostatistical Software Library and User's Guide, Oxford University Press, New York, 2nd edition (1998)

[17] DiPerna RJ, Measure-Valued Solutions to Conservation Laws, Archive for Rational Mechanics and Analysis, Vol. 88, No. 3, p. 223-263 (1985)

[18] Edwards MG, Higher Dimensional Wave Oriented Schemes with Minimal Cross-Wind Diffusion, SPE 79689 (2000)

[19] E Weinan, Linear and Nonlinear Transport Equations, Communications on Pure and Applied Mathematics, Vol. XLV, p. 301-326 (1992)

[20] Efendiev Y, Durlofsky IJ, Lee SH, Modeling of Subgrid Effects in Coarse-Scale Simulations of Transport in Heterogeneous Media, Water Resources Research, Vol. 36, No. 8, p. 2031-2041 (2000)

[21] Efendiev Y, Durlofsky IJ, Numerical Modeling of Subgrid Heterogeneity in Two-Phase Flow, Water Resources Research, Vol. 38, No. 8, Art. No. 1128 (2002)

[22] Efendiev Y, Ginting V, Hou T, Ewing R, Accurate Multiscale Finite Element Methods for Two-Phase Flow Simulations, Journal of Computational Physics (submitted 2005)

[23] Efendiev Y, Hou TY, Wu X, Convergence of a Nonconforming Multiscale Finite Element Method, SIAM Journal of Numerical Analysis, Vol. 37, No. 3, p. 888-910 (2000)

[24] Efendiev Y, Popov B, On Homogenization of Nonlinear Hyperbolic Equations, Communications on Pure and Applied Analysis, Vol. 4, No. 2, p. 295-309 (2005)

[25] Gerritsen M, Durlofsky LJ, Modeling Fluid Flow in Oil Reservoirs, Annual Review of Fluid Mechanics, Vol. 37, p. 211-238 (2005)

[26] Higgins RV, Leighton AJ, A Computer Method to Calculate Two-Phase Flow in Any Irregularly Bounded Porous Medium, Journal of Petroleum Technology, Vol. 14, p. 679-683 (1962)

[27] Hornung U, Homogenization and Porous Media, Springer-Verlag New York (1997)

[28] Hou TY, Wu XH, A Multiscale Finite Element Method for Elliptic Problems in Composite materials and porous media, Journal of Computational Physics, Vol. 134, p. 913-943 (1997)

[29] Hou TY, Xin X, SIAM Journal on Applied Mathematics, Vol. 52, No. 1, p. 34-45 (1992)

[30] Hou TY, Yang DP, Wang K, Homogenization of Incompressible Euler Equations, Journal of Computational Physics, Vol. 22 (2004)

[31] Hou TY, Yang DP, Ran H, Multiscale Analysis in Langrangian Formulation for the 2-D Incompressible Euler Equation, Discrete and Continuous Dynamical Systems, Vol. 12, No. 0 (2005)

[32] Huang WZ, Ren RJ, Russel RD, Moving Mesh Partial Differential Equations (MMPDEs) Based on the Equidistribution Principle, SIAM Journal of Numerical Analysis, Vol. 31, No. 3, p.709-730 (2004)

[33] Jenny P, Lee SH, Tchelepi HA, Multi-Scale Finite-Volume Method for Elliptic Problems in Subsurface Flow Simulation, Journal of Computational Physics, Vol. 187, p. 47-67 (2003)

[34] Kružkov SN, First-Order Quasilinear Equations in Several Variables, Mathematics of the USSR-Sbornik, Vol. 10, No. 2, p. 217-243 (1970)

[35] LeFloch PG, Entropy Weak Solutions to Nonlinear Hyperbolic Systems under Nonconservative Form, Communications in Partial Differential Equations, 13(6), p. 669-727 (1988)

[36] Leveque RJ, Finite Volume Methods for Hyperbolic Problems, Cambridge University Press (2002)

[37] Li R, Tang T, Zhang P, Moving Mesh Methods in Multiple Dimensions based on Harmonic Maps, Journal of Computational Physics, Vol. 170, p. 562-588 (2001)

[38] Muskat M, Flow of Homogeneous Fluids, International Human Resources Development Corporation, 137 Newbury Street, Boston MA 02116 (1937, 1982)

[39] Muskat M, Meres MW, The Flow of Heterogeneous Fluids through Porous Media, Physics 7 (Sept) p. 346-363 (1936)

[40] McLaughlin SW, Papanicolaou G, Pironneau O, Convection of Microstructure and Related Problems, SIAM Journal of Applied Mathematics, Vol. 45, p. 780-797 (1985)

[41] Nguetseng G, A General Convergence Result for a Functional Related to the Theory of Homogenization, SIAM Journal of Mathematical Analysis, Vol. 20, p. 2636-2647 (1987)

[42] Osher S, Riemann Solvers, the Entropy Condition, and Difference Approximations, SIAM Journal of Numerical Analysis, Vol. 21, p. 217-235 (1984)

[43] Papanicolaou GC, Kohler W, Asymptotic Theory of Mixing Stochastic Ordinary Differential Equations, Communications of Pure and Applied Mathematics, Vol. 27, p. 641-668 (1974)

[44] Pollock DW, Semianalytical Computation of Path Lines for Finite-Difference Models, Ground Water, Vol. 26, No. 2, p. 743-750 (1988)

[45] Shafer JM, Reverse Pathline Calculation of Time-Related Capture Zones in Nonuniform Flow, Ground Water, Vol. 25, No. 3, p. 283-289 (1987)

[46] Sibson R, A Vector Identity for the Dirichlet Tessellation, Mathematical Proceedings of the Cambridge Philosophical Society, Vol. 87, p. 151-155 (1980)

[47] Strikwerda JC, Finite Difference Schemes and Partial Differential Equations, Chapman and Hall (1989)

[48] Tan ZJ, Zhang ZR, Huang YQ, Tang T, Moving Mesh Methods with Locally Varying Time Steps, Journal of Computational Physics, Vol. 200, No. 1, p. 347-367 (2004)

[49] Tartar L, Nonlocal Effects Induced by Homogenization, PDE and Calculus of Variations, Essays in Honor of Ennio de Giorgi, Colombini F et al eds., Birkhäuser, Boston, p. 925-938 (1989)

[50] Thiele MR, Batycky RP, Blunt MJ, Orr FM, Simulating Flow in Heterogeneous Media Using Streamtubes and Streamlines, SPE Reservoir Engineering, Vol. 10, No. 1, p. 5-12 (1996)

[51] van Leer B, Computational Methods in Applied Sciences and Engineering VI, edited by R. Glowinski and J-L Lions, North Holland, Amsterdam, p. 493 (1984)

[52] Wen XH, Durlofsky LJ, Edwards MG, Upscaling of Channel Systems in Two Dimensions Using flow-based Grids, Transport in Porous Media, Vol. 51, p. 343-366 (2003)

[53] Westhead A, Upscaling Two-Phase Flows in Porous Media, Caltech Thesis (2005)

Appendix

Proof that $f_S\overline{S_p}$ depends weakly on time

In this section we justify that $f_S(\overline{S}(P(p,\tau),\psi,\tau))\overline{S}_p(P(p,\tau),\psi,\tau)$ does not vary much along the streamlines so that we can take it outside the time integral of the macrodispersion which is written out below

$$\overline{\tilde{v_0}' f_S(\overline{S})S'} = -\int_0^t \overline{\tilde{v_0}' f_S(\overline{S})\tilde{v_0}(P(p,\tau),\psi) f_S(\overline{S}(P(p,\tau),\psi,\tau))\overline{S}_p(P(p,\tau),\psi,\tau)}d\tau.$$

By "weakly in time" we mean that taking the term $f_S(\overline{S}(P(p,\tau),\psi,\tau))\overline{S}_p(P(p,\tau),\psi,\tau)$ outside the time integration of the macrodispersion term introduces an error that is third-order in fluctuating quantities. To prove it we follow an idea in [**14**] and write the macrodispersion in terms of $G(p,P(p,\alpha)) = \tilde{v_0}'\tilde{v_0}(P(p,\alpha),\psi)$

$$\overline{\tilde{v_0}' f_S(\overline{S})S'} = -f_S(\overline{S})\overline{\int_0^t \frac{d}{d\tau}\int_0^\tau G(p,P(p,\alpha))d\alpha \frac{\partial}{\partial p} f(\overline{S}(P(p,\tau),\psi,\tau))d\tau}.$$

We integrate by parts to find

$$\overline{\tilde{v_0}' f_S(\overline{S})S'} = -f_S(\overline{S})\overline{\int_0^t G(p,P(p,\tau))d\tau \frac{\partial}{\partial p} f(\overline{S}(P(p,t),\psi,t))} + \tag{6.0.1}$$
$$+ f_S(\overline{S})\overline{\int_0^t \int_0^\tau G(p,P(p,\alpha))d\alpha \frac{\partial}{\partial p}\frac{\partial}{\partial \tau} f(\overline{S}(P(p,\tau),\psi,\tau))d\tau}.$$

The first term on the right hand side of (6.0.1) is what we model the macrodispersion with. We must show that the second term is much smaller than the first. If we solve the equation for the average saturation (4.4.2) along the coarse streamlines we obtain

$$\frac{d}{d\tau}\overline{S}(P(p,\tau),\psi,\tau) = \overline{\tilde{v_0}'(f_S(\overline{S})S')_p}.$$

This term is second-order in fluctuating quantities, whereas $f(\overline{S}(P(p,t),\psi,t))$ is zeroth order. Together with the term $\int_0^\tau G(p,P(p,\alpha))d\alpha$ they make the second term of (6.0.1) third-order in fluctuating quantities. Then we obtain the following expression for the macrodisperion

$$\overline{\tilde{v_0}' f_S(\overline{S})S'} = -\int_0^t \overline{\tilde{v_0}'\tilde{v_0}(P(p,\tau),\psi)}d\tau f_S(\overline{S})^2\overline{S}_p.$$

Index