Advances in the Discontinuous Galerkin Method: Hybrid Schemes and Applications to the Reactive Infiltration Instability in an Upwelling Compacting Mantle

by

Alan R. Schiemenz

B.S., Baylor University, Texas, 2004

Sc.M., Brown University, Rhode Island, 2005

A dissertation submitted in partial fulfillment of the
requirements for the degree of Doctor of Philosophy
in the Division of Applied Mathematics at Brown University

PROVIDENCE, RHODE ISLAND

May 2009

Abstract of "Advances in the Discontinuous Galerkin Method: Hybrid Schemes and Applications to the Reactive Infiltration Instability in an Upwelling Compacting Mantle" by Alan R. Schiemenz, Brown University, May 2009

Abstract

High-order methods are emerging in the scientific computing community as superior alternatives to the classical finite difference, finite volume, and continuous finite element methods. The discontinuous Galerkin (DG) method in particular combines many of the positive features of all of these methods. This thesis presents two projects involving the DG method.

First, a Hybrid scheme is presented, which implements DG areas where the solution is considered smooth, while dropping the order of the scheme elsewhere and implementing a finite volume scheme with high-order, non-oscillatory solution reconstructions suitable for unstructured mesh. Two such reconstructions from the ENO class are considered in the Hybrid. Successful numerical results are presented for nonlinear systems of conservation laws in one dimension.

Second, the high-order discontinuous Galerkin and Fourier spectral methods are applied to an application modeling three-phase fluid flow through a porous medium, undergoing solid-fluid reaction due to the reactive infiltration instability (RII). This model incorporates a solid upwelling term and an equation to track the abundance of the reacting mineral orthopyroxene (opx). After validating the numerical discretization, results are given that provide new insight into the formation of melt channels in the Earth's mantle. Mantle heterogeneities are observed to be one catalyst for the development of melt channels, and the dissolution of opx produces interesting bifurcations in the melt channels. An alternative formulation is considered where the mass transfer rate relative to velocity is taken to be infinitely large. In this setting, the stiffest terms are removed, greatly reducing the cost of time integration.

This dissertation by Alan R. Schiemenz is accepted in its present form by the Division of Applied Mathematics as satisfying the dissertation requirement for the degree of Doctor of Philosophy.

Date_____ _____

Jan S. Hesthaven, Ph.D., Advisor

Recommended to the Graduate Council

Date_____ _____

Chi-Wang Shu, Ph.D., Reader

Date_____ _____

Edgar M. Parmentier, Ph.D., Reader

Approved by the Graduate Council

Date_____ _____

Sheila Bonde, Dean of the Graduate School

The Vita of Alan R. Schiemenz

Born on August 12, 1983 in Waco, Texas.

Education

- Sc. M. in Applied Mathematics, Brown University, May 2005.
- B.S. in Mathematics, Baylor University, Waco, Texas, May 2004.

Publications/Preprints

- Schiemenz, A., Hesse, M., Liang, Y., Parmentier, E., and Hesthaven, J., *Reactive Infiltration Instability in an Upwelling Compacting Mantle: Linear Stability Analysis*, in progress.
- Schiemenz, A., Hesse, M., Liang, Y., Parmentier, E., and Hesthaven, J., *Reactive Infiltration Instability in an Upwelling Compacting Mantle: Nonlinear Numerical Simulations*, in progress.
- Schiemenz, A. and Robinson, A., *CSRI Summer Proceedings 2007 - The Computer Science Research Institute at Sandia National Laboratories*, edited by M. L. Parks and S. S. Collis, http://www.cs.sandia.gov/CSRI/Proceedings/CSRI2007.pdf, SAND2007-7977, 2007, pp. 231-241, Energy Based Magnetic Forces Computation using Automatic Differentiation.

Research Interests

- Numerical solutions to partial differential equations.
- Finite element methods, particularly the discontinuous Galerkin (DG) method.
- High-order numerical methods for fluid flow through porous media.
- High-order limiting and essentially non-oscillatory (ENO) reconstruction methods.

Academic Experience

- Teaching Assistant: *Methods of Applied Mathematics, I*, Division of Applied Mathematics, Brown University, Fall 2005 & 2008.
- Grading Assistant: *Numerical Solutions to Partial Differential Equations, II*, Division of Applied Mathematics, Brown University, Spring 2008.
- Private Mathematics Tutor, January 2007 - May 2009.
- Math Resource Center, Brown University, August 2006 - May 2007.

Awards and Honors

- Student Travel Support Grant, SIAM conference on mathematical & computational issues in the geosciences, June 2009, Leipzig, Germany.
- Dissertation Fellowship, Brown University, Spring 2009. Graduate Student Fellowship, Division of Applied Mathematics, Brown University, 2003-2004.
- Phi Beta Kappa, inducted 2004 into the Zeta of Texas chapter.
- Outstanding Mathematics Student, 2004, Baylor University.

Presentations

- SIAM Conference on Mathematical & Computational Issues in the Geosciences, June, 2009. *High-order Methods for the Reactive Infiltration Instability in an Upwelling Compacting Mantle.*
- American Geophysical Union Fall Meeting, December, 2008. *A Possible Link between Mantle Heterogeneity and Dunite Channel Distribution in an Upwelling Mantle: A Numerical Study.*
- Sandia Student Symposium, August 2007. *Improving Magnetic Energy Force Discretization in ALEGRA.*

- Brown University, Division of Applied Mathematics "Pizza Seminar" Series,May 2007. *Discontinuous Galerkin Method: Introduction, Analysis and use in Hybrid Schemes.*

Acknowledgments

Foremost I would like to thank the Lord for His providential plan in my life. When I arrived at Brown in the fall of 2004 I set foot in a new world, with little understanding of what awaited. In the five years since much has been revealed, and through the enduring experience of graduate school the Lord has granted me the opportunity for growth both in my professional endeavors and my character.

I would like to sincerely thank my advisor, Professor Jan S. Hesthaven, for his willingness to take me as a student. He has invested a great amount of his time and effort in my education, always providing friendly mentorship and encouragement.

My thanks go out to my colleagues, Dr. Marc Hesse, and Professors Jan Liang and Marc Parmentier. They have introduced me to the field of computational geosciences. Moving to the application realm of numerical analysis has greatly shaped my professional interests.

I would like to express my appreciation to my friends and fellow graduate students, Akil Naryan, Andreas Kloeckner, Michael Lamar, Ravi Srinivasan and Kenny Chowdhary. They have provided me with great friendship and technical assistance over the years.

Finally, I am forever indebted to my family. I thank my parents for their unwavering support in often-difficult times during my graduate career. I thank my wife for her grace and companionship during this time in my life. Without the encouragement and love of my family this accomplishment would have been impossible.

Contents

List of Tables

List of Figures

Chapter 1

Discontinuous Galerkin Methods

The two projects presented in this thesis build upon the foundation of the discontinuous Galerkin (DG) finite element method. The method will be presented in this chapter, with discussion given to implementation-oriented concerns such as modal and nodal representations and stability for nonlinear problems. A brief discussion will be given for explicit Runge-Kutta time integration methods. Finally, we will discuss the lowest-order implementation of DG, the finite volume method.

1.1 History

The discontinuous Galerkin method is a popularly emerging choice for the numerical solution to partial differential equations. The first DG method was introduced in 1973 [47] as a means of solving the steady-state neutron transport equation. Major developments in the DG method were introduced in [12] with the combination of explicit, nonlinearly stable Runge-Kutta methods [50] with an improved slope limiter [49], allowing formal accuracy in regions where the solution is smooth, sharp shock resolution, and convergence to the entropy solution. Further progress was made with the extension to high-order accuracy in [13] and one-dimensional systems in [11]. In [10], these results were extended for multidimensional scalar problems, and finally

to systems in [14] with the introduction of a special class of slope limiters.

1.2 Motivation

DG combines many attractive features of the classical finite element (FE), finite volume (FV), and finite difference (FD) methods. In short, DG schemes, with a polynomial solution and test function basis, appear similar in presentation to continuous finite element method ("Galerkin"). However, there are no continuity requirements on the solution at element interfaces ("discontinuous"), which more resembles the nature of the finite volume method. Indeed, the lowest order DG scheme is equivalent to FV, as the evolved degree of freedom is a cell average.

Generally, however, one chooses to implement DG for its ability to provide a high-order solution, which contrasts with the inherently low-order FV, and its freedom to be implemented on unstructured grids, which contrasts with FD. DG is ideally suited for conservation laws and problems with propagating waves, as the discontinuous solution representation at cell interfaces decouples the global solution into a collection of local ones. Therefore, rather than having large global mass and stiffness matrix operators, we have local operators that may be tailored specifically to the physics of the problem at hand. This is in contrast to the continuous FE method.

One obstacle that DG faces, however, is the well-known order barrier theorem from Godunov which states that high-order schemes are necessarily non-monotonic. For smooth solutions this poses no problems, but in the presence of strong discontinuities, the numerical solution will contain non-physical oscillations. Techniques such as slope limiting have been used to resolve this, but generally require fine-tuning of solution and/or mesh-dependent parameters in order to avoid reducing the order of the scheme in smooth regions. More robust limiter choices such as WENO [46, 44] and Hermite WENO [43, 45] shows great promise, but are less suitable for unstructured grids. In Chapter 3 a hybrid scheme, which switches between the dis-

continuous Galerkin method and a finite volume method with high-order function reconstruction for the evaluation of numerical fluxes, is presented. In this work positive research results (high-order, non-oscillatory) are given for the numerical solution to one-dimensional conservation laws on unstructured grids.

1.3 Basic Formulation

In this section we present the basic formulation of the discontinuous Galerkin method scheme for scalar conservation laws. These are generally written as

$$\begin{aligned} \frac{\partial u\left(x,t\right)}{\partial t}+\nabla\cdot f\left(u(x,t),x,t\right)=0,\ x\in\Omega\subset\Re^{n} \\ u\left(x,t\right)=g\left(x,t\right),\ x\in\partial\Omega \\ u\left(x,0\right)=u_{0}(x),\ x\in\Omega \end{aligned} \tag{1.1}$$

where the flux function f is given, and boundary conditions g are available to ensure well-posedness of the PDE.

Tesselating the domain into elements $\Omega=\bigcup_{k}T^{k}$, locally, the N-order DG solution to (1.1) is chosen from the finite polynomial basis

$$V_{h}^{k}=\left\{\phi^{k}:\phi^{k}\in P^{N}(T^{k}),k=1,2,...,K\right\}, \tag{1.2}$$

where $P^{N}(T^{k})$ is the set of polynomials of degree N with support in element T^{k}. We define the dimension of V_{h}^{k} to be N_{p}, and for general tetrahedral elements in n-dimensional space, the scaling between N and N_{p} is

$$N_{p}=\left(\begin{array}{c} n+N \\ N \end{array}\right)\sim\frac{N^{n}}{n!} \tag{1.3}$$

Returning to (1.2), the basis V_{h}^{k} is selected as both the solution basis and the

test function basis, as is standard for a Galerkin scheme. There are no continuity requirements on the solution across element boundaries ∂T^k. For simplicity we assume that the same basis V_h^k is used for each element, although we may certainly vary this, i.e., adaptively choosing N based upon the expected smoothness of the solution. Locally, then, the solution may be represented as either a linear combination of basis functions

$$u_h^k(x,t) = \sum_{j=1}^{N_p} \hat{u}_j^k(t)\phi_j^k(x), x \in T^k \tag{1.4}$$

or equivalently as a linear combination of nodal values

$$u_h^k(x,t) = \sum_{j=1}^{N_p} u_h^k(x_j^k,t)l_j^k(x), x \in T^k. \tag{1.5}$$

Here, a set of gridpoints $\left\{x_j^k\right\}_{j=1}^{N_p}$ has been introduced along with the corresponding Lagrange interpolating polynomials l_j^k.

For general tetrahedral mesh, any two elements may be related to one another via a linear transformation, and so for implementation purposes we may perform all computations on a reference element T^E with interpolation points ξ_j and then map back to T^k. Thus, all operators are local, with only the need to store the Jacobians of each element's transformation to T^E.

On the reference element, then, the two solution representations (1.4) and (1.5) become (suppressing time dependence)

$$\begin{aligned} u_h^E(\xi) &= \sum_{j=1}^{N_p} \hat{u}_j \phi_j(\xi) \\ &= \sum_{j=1}^{N_p} u(\xi_j) l_j(\xi) \end{aligned} \tag{1.6}$$

The degrees of freedom $\hat{u}_j$ and $u(\xi_j)$ are linked via the Vandermonde matrix

$$\mathcal{V}\hat{\mathbf{u}} = \mathbf{u} \tag{1.7}$$

where

$$\mathcal{V}_{ij} = \phi_{j-1}(\xi_i), \quad \hat{\mathbf{u}}_i = \hat{u}_i, \quad \mathbf{u}_i = u(\xi_i). \tag{1.8}$$

Practically speaking, it is easier to perform mathematics on the modal representation (1.4) if one chooses orthonormal ϕ_j^k. However, it is easier to visualize the scheme (and thus, the solution) by considering the nodal representation (1.5).

The Vandermonde matrix is typically found embedded in other operators in the scheme (e.g. the mass and stiffness matrices), so that the solution remains in nodal form while computations are performed in modal form.

The choice of which basis functions ϕ_j^k or which interpolation values ξ_j^k to use are of significance when one considers issues such as matrix conditioning; see [26, 59] and the references therein for more details. Typically for ϕ_j^k we use the orthogonal Legendre polynomials, a special subset of the Jacobi polynomials, with corresponding Gauss-Lobatto interpolation points ξ_j^k (See Sec. 4.4 for more discussion). In the latter case the basis functions become the Lagrange interpolating polynomials l_j^k. Greater discussion is given to the derivation of the Gauss-Lobatto points in Sec. 4.4.2.

Having defined the solution space, we proceed with defining the scheme by considering the variational form of (1.1). Multiplication by test function l_j^k and integration over T^k yields

$$\int_{T^k} \left(\frac{\partial u_h^k}{\partial t} + \nabla \cdot f_h^k \right) l_j^k dx = 0, \quad j = 1, 2, ..., N_p \tag{1.9}$$

where the discretizations u_h^k, f_h^k have now been introduced as approximations to u and f on element k. Using the divergence theorem we then integrate by parts to

arrive at the residual statement

$$\int_{T^k} \left(\frac{\partial u_h^k}{\partial t} l_j^k - f_h^k \cdot \nabla l_j^k \right) dx = - \oint_{\partial T^k} \hat{n} \cdot \left(f_h^k \right)^* l_j^k dx, \quad j = 1, 2, ..., N_p. \tag{1.10}$$

Alternatively, we may integrate by parts again and have

$$\int_{T^k} \left(\frac{\partial u_h^k}{\partial t} - \nabla \cdot f_h^k \right) l_j^k dx = - \oint_{\partial T^k} \hat{n} \cdot \left[f_h^k - \left(f_h^k \right)^* \right] l_j^k dx, \quad j = 1, 2, ..., N_p. \tag{1.11}$$

Equations (1.10) and (1.11) are the weak and strong forms, respectively, of the discontinuous Galerkin method. The two forms are mathematically equivalent but differ in implementation. Noting that the numerical solution u_h^k is discontinuous across element boundaries ∂T^k, we must introduce a new approximation $\left(f_h^k \right)^*$, the numerical flux, to compute the right hand side of these equations. Whereas the f_h^k terms and its derivatives are computed exclusively on the element T^k, the numerical flux $\left(f_h^k \right)^*$ depends on values taken both from T^k and its neighbors. As a result, the local solutions become coupled, allowing for the global transfer of information. Further discussion concerning the numerical flux can be found in Section 1.6.1. In Sec.4.4.4 an implementation of the strong form is given for a system of conservation laws.

Having now arrived at a spatial discretization (1.10) or (1.11), we may integrate the solution in time with a suitable time-stepping scheme. Next, we discuss the explicit Runge-Kutta method, a special class of such schemes.

1.4 Explicit Runge-Kutta Time Integration

The equations resulting from (1.10) and (1.11) reflect a method of lines approach, where the spatial discretization of (1.1) reduces the partial differential equation to a system of ordinary differential equations. This results in a semi-discrete scheme of

the form

$$\frac{du_h}{dt} = \mathcal{L}_h(u_h, t) \tag{1.12}$$

for some corresponding operator $\mathcal{L}_h$. We then advance the solution with a suitable time integration scheme. One such method comes from the class of explicit Runge-Kutta (ERK) schemes.

General $m-$order ERK schemes [52] advance the solution $u_h^n = u_h(\cdot, t^n)$ in time to $u_h^{n+1} = u_h(\cdot, t^{n+1}) = u_h(\cdot, t^n + \Delta t)$ by evaluating the operator $\mathcal{L}_h$ in m stages,

$$\begin{aligned} v^{(0)} &= u_h^n \\ v^{(i)} &= \sum_{k=0}^{i-1} \left(\alpha_{i,k} v^{(k)} + \Delta t \beta_{i,k} \mathcal{L}_h \left(v^{(k)}, t^n + \Delta t^{(k)} \right) \right), \qquad i = 1, ..., m, \\ v^{(m)} &= u^{n+1}. \end{aligned} \tag{1.13}$$

The $m = 2, 3$ and 4 versions of the classic ERK (1.13) are popular choices for time integration, but come with the drawback of needing to store each of the intermediate stages $v^{(i)}$. An alternative to the fourth order scheme is the low-storage version LSERK [7],

$$\begin{aligned} & p^{(0)} = u_h^n, \\ & i \in [1, ..., 5] : \begin{cases} v^{(i)} &= a_i v^{(i-1)} + \Delta t \mathcal{L}_h \left(p^{(i-1)}, t^n + c_i \Delta t \right), \\ p^{(i)} &= p^{(i-1)} + b_i v^{(i)} \end{cases} \\ & u_h^{n+1} = p^{(5)} \end{aligned} \tag{1.14}$$

Although the fourth-order LSERK requires an additional stage of computation, there is only one additional storage level $v^{(i)}$, thereby significantly decreasing the storage cost. Moreover, LSERK potentially allows for a larger stable timestep Δ_t than its classic ERK counterpart; see [28] for details.

Runge-Kutta methods are the most popular selection for time-integration. For more choices, see [6, 23, 24].

1.5 Problems With Nonlinear Problems

Having now walked through a basic discontinuous Galerkin scheme, we turn to focus on one potential problem that may arise with a nodal implementation. It can be proven [29] that the discontinuous Galerkin solution to (1.1) converges to the entropy solution if f is convex. Practically speaking, however, caution should be urged in implementation. If the solution is represented by storing its nodal values as in (1.5), then we can evaluate the flux as

$$f_h^k = \sum_{j=1}^{N_p} f(u_h^k(x_j^k, t)) l_j^k(x), \tag{1.15}$$

that is, by simply reading u_h at the grid points and evaluating for f_h. This is an easy and straightforward way to compute the flux, but only accurate if f itself is able to represented by the basis chosen to represent u_h. For example, if $f(u) = u^4$, $u_h \in P^N$, then $f_h \in P^{4N}$, and the representation (1.15) will have insufficient number of gridpoints to completely represent the solution. Thus, we would incur aliasing errors that could potentially lead to instability in the numerical solution when evaluating stiffness terms such as the integral $\int_{T^k} (\nabla \cdot f_h) \, l_j^k dx$. Obviously, if we were to increase the order of approximation N, the increased resolution would eventually be able to sufficiently capture f_h. While this is a robust way to solve the problem it will certainly increase the computational cost and often is not practical. Further discussion of this can be found in [22, 27].

A different solution is to add artificial dissipation to the problem as a means of filtering out the unresolvable, highest-order modes of the solution. Typically, this involves solving a modified form of (1.1) in which we add additional terms of the form

$\frac{\partial^{2s} u}{\partial x^{2s}}$ for some (even) integer $2s$. Instead of worrying about the operators involved with these terms and how they would be incorporated into the scheme, we may simply post-process the solution to achieve the goal at hand:

$$u_h(\xi) \to F u_h(\xi) = \sum_{j=1}^{N_p} \sigma\left(\frac{j-1}{N}\right) \hat{u}_j \phi_j \tag{1.16}$$

where the exponential filtering function σ is defined as

$$\sigma(\eta) = \begin{cases} 1, & 0 \leq \eta \leq \eta_c = \frac{N_c}{N} \\ \exp\left(-\alpha \left(\left(\eta - \eta_c\right) / \left(1 - \eta_c\right)\right)^s\right), & \eta_c \leq \eta \leq 1. \end{cases} \tag{1.17}$$

From (1.16)-(1.17) we can see that the highest order modes in the solution are dampened, depending upon the parameters chosen. The parameter N_c acts as a cutoff frequency whereby lower-order modes are unfiltered. α is a dissipation parameter, usually taken to be on the order of $\alpha = -\log(\epsilon_M)$, where ϵ_M is machine precision. Whereas α tends to remain fixed, the strength parameter s adds a dimension of variability; a good starting point for this value is $s = 16$. Higher N_c and s correspond to less filtering.

For the second project in this thesis presented in Chapter 4, which presents high-order methods for the reaction infiltration instability, filtering is used to dissipate the aliasing errors introduced from the nonlinearity in the equations. The numerical schemes in this work implement a Fourier spectral method; the exponential filter presented may also be applied in this context as well. Details of the implementation can be found in [27].

1.6 Introduction to Finite Volume Schemes

In the lowest order ($N = 0$) setting, the discontinuous Galerkin method reduces to a finite volume scheme. In this section we will step through a one-dimensional imple-

mentation, providing further discussion of the numerical flux which was previously introduced in the general formulation of the discontinuous Galerkin method.

We consider the one-dimensional scalar conservation law

$$\begin{aligned} &\frac{\partial u(x,t)}{\partial t} + \frac{\partial f(u)}{\partial x} = 0,\ x \in \Omega \subset \mathbb{R}, t > 0 \\ &u(x,0) = u_0(x) \end{aligned} \tag{1.18}$$

subject to suitable boundary conditions so as to guarantee well-posedness of the problem. We assume a domain $\Omega = \bigcup_{k=1}^{K} T^k$, where the elements T^k are the intervals

$$T^k = \left[x^{k-1/2}, x^{k+1/2}\right], \tag{1.19}$$

so that $\left|T^k\right| \equiv x^{k+1/2} - x^{k-1/2}$. Using this notation, we may denote f on cell boundaries, $f^{k\pm1/2} = f\left(x^{k\pm1/2}\right)$.

The finite volume scheme is derived by integrating (1.18) over each interval T^k,

$$\frac{d\bar{u}^k}{dt} = -\frac{1}{|T^k|}\left(f^{k+1/2} - f^{k-1/2}\right) \tag{1.20}$$

where we have introduced the notation for the cell average $\bar{u}^k$,

$$\bar{u}^k(t) \equiv \frac{1}{|T^k|}\int_{x^{k-1/2}}^{x^{k+1/2}} u(z,t)dz. \tag{1.21}$$

These averages are the fundamental degrees of freedom to be solved for. The solution is therefore discontinuous across cell interfaces $x^{k\pm1/2}$, and so we must approximate the flux f in these locations. This approximation, the so-called *numerical flux* is denoted

$$\left(f^{k+1/2}\right)^* = h\left((u-)^{k+1/2}, (u+)^{k+1/2}\right) \tag{1.22}$$

where the notation $(u-)^{k+1/2}$ and $(u+)^{k+1/2}$ refers to left- and right-sided approximations of u, with respect to the interface between elements T^k and T^{k+1}.

1.6.1 Numerical Flux

The numerical flux function $h = h(a, b)$ introduced in Eq. (1.22) must satisfy the following three properties:

- h is Lipschitz continuous with respect to both arguments
- h is nondecreasing with respect to its first argument and nonincreasing with respect to its second argument
- h is consistent with the original flux f, that is, $h(a, a) = f(a)$

There are a great number of numerical flux functions that may suffice for a given problem. A comprehensive listing of these can be found in [36]. Perhaps the easiest and most stable (dissipative) choice is the Lax-Friedrichs flux

$$h^{LF}(a, b) = \frac{1}{2}\left[f(a) + f(b) - C\,(b - a)\right] \tag{1.23}$$

where the constant C is

$$C = \max_{u} \left|\frac{df}{du}\right|. \tag{1.24}$$

The maximum may be applied over the entire domain Ω (global) or in a more restricted range (local), such as in small neighborhood of the element interface. In this case C will vary throughout the domain. The global Lax Friedrichs flux provides a more dissipative scheme.

An alternative approach to (1.23) adds a control $0 \leq \alpha \leq 1$ to the dissipative term by taking

$$h(a, b) = \frac{1}{2}\left[f(a) + f(b) - C\,(1 - \alpha)\,(b - a)\right]. \tag{1.25}$$

In this way, choosing $\alpha = 0$ recovers (1.23) while choosing $\alpha = 1$ results in a purely central flux with no dissipation. The latter is the preferred choice when dealing with problems with no preferred direction of propagation, e.g. the diffusion equation.

Returning to the numerical scheme (1.20), then, we require a reconstruction of the values $u(x^{k\pm1/2})$ to approximate $\left(f^{k\pm1/2}\right)^*$. Once this is known we may evolve the solution in time utilizing an ERK method, as presented in Section 1.4.

1.7 Conclusion

This chapter has introduced the discontinuous Galerkin finite element scheme, with additional discussion of its lowest-order ($N = 0$) version, the finite volume scheme. Essential for both formulations is the numerical flux. For the $N > 0$ discontinuous Galerkin method the numerical flux may be readily computed by considering the solution values on all sides of the interface in question. For the finite volume scheme, however, this information is not available. In the next chapter we will detail solution reconstruction algorithms which provide the needed quantities.

Part I

Hybrid Discontinuous Galerkin Methods

Chapter 2

ENO-Type Reconstructions

In both the discontinuous Galerkin method as well as the finite volume method, the discontinuous solution is connected to its neighbors via the numerical flux. The numerical flux is computed via an approximate Riemann solver, considering the values of the multiply-defined solution across element boundaries. For the discontinuous Galerkin method of order 1 and greater, these values are naturally available due to the polynomial basis which represents a continuous local expression of the solution. However, for finite volume schemes the fundamental degrees of freedom used are merely the cell-averages. The computation of the numerical flux, therefore, requires a reconstruction procedure whereby one uses these degrees of freedom to compute an approximation of the solution at element boundaries.

ENO, or essentially non-oscillatory methods, first presented in [25], may be used to provide an oscillation-free reconstruction capable of preserving high-order features. The main idea of ENO is to pick the "best" solution reconstruction among those computed from a collection differing stencils. WENO [37, 30] methods improve upon this idea by instead considering a weighted sum of reconstructions across *all* stencils, thereby improving the overall order of accuracy. The particular weights to use are computed via a nonlinear smoothness indicating functional of the cell-averaged degrees of freedom, with nonsmooth reconstructions receiving weights on

the order of the truncation error in order to attain the ENO property.

The main difficulty involved in the reconstruction process is that of maintaining accuracy. In particular, high-order features of a solution may be incorrectly computed without a suitably high-order reconstruction. At the other end of the spectrum, solutions with strong shocks may suffer non-physical oscillations or excessive smearing due to poor reconstruction. Moreover, standard WENO schemes [37, 51] are inherently one-dimensional in nature, performing reconstructions direction-by-direction on structured mesh. For the more complicated case of an unstructured mesh in multiple dimensions, ENO-type reconstructions are hindered by the need for potentially large stencils and complicated stencil-searching algorithms to detect the best stencil(s) from which to build the solution. Recent developments have included Hermite WENO reconstructions [43, 45], which include both the cell average and first derivatives as the degrees of freedom. These show promise in reducing the stencil size, but still carry the requirement of a structured mesh. A Hermite WENO scheme for unstructured mesh is given in [38], but is as yet only presented for linear reconstructions.

In this chapter, we will first discuss the general attributes of ENO-type reconstructions. Next, we will detail the implementation of two particular high-order ENO-type reconstruction algorithms, QENO and DK, proposed in related forms in [41] and [17, 18], respectively, will be discussed. The former resembles an ENO scheme by reconstructing a single polynomial while the latter is a WENO scheme, with multiple reconstructions weighted in a nonlinear fashion. Both involve solving least-squares problems, which allows for grace in the stencil-searching procedure to seek out smooth data from which to reconstruct the solution.

2.1 Properties of the ENO Class

The function reconstruction procedure is as follows: given a stencil of cell averages

$$\bar{S}^k = \left\{ \bar{u}^{k-p}, \bar{u}^{k-p+1}, ..., \bar{u}^{k+q-1}, \bar{u}^{k+q} \right\}, \tag{2.1}$$

we wish to build an approximating polynomial used to evaluate u on cell boundaries, $u^{k\pm 1/2}$. Perhaps the simplest way to do this would be via an averaging, i.e.

$$u^{k+1/2} = \frac{1}{2}\left(\bar{u}^k + \bar{u}^{k+1/2}\right). \tag{2.2}$$

With larger stencils, one could expand this interpolation to higher order accuracy. Unfortunately, as is well-known in function interpolation theory, such reconstructions of order two or greater will necessarily produce oscillatory data should the stencil include discontinuous solutions. ENO-type reconstructions are a special class of reconstruction algorithms that produce high-order, (essentially) non-oscillatory reconstructions. We seek a reconstruction of this class with the following properties:

- **Conservation of mean**: We must preserve the fundamental degrees of freedom (1.21).

- **Accurate representation of smooth functions**: To maintain accuracy for smooth functions it is required that for any smooth function $u(x)$, the corresponding local reconstruction $R_i(x - x_i)$ will satisfy

$$R_i(x - x_i) = u(x) + O(\Delta x^{N+1}). \tag{2.3}$$

- **Compact support**: One very attractive property of the discontinuous Galerkin scheme is that locally, the solution only depends upon its immediate neighbors. Unfortunately this is not necessarily the case with finite volume schemes as the reconstruction of function values at cell boundaries may involve the need to

consider values from a wide stencil of neighbors in addition to the local element itself. The construction of a one-dimensional polynomial of (high-order) degree N, for example, requires at least $N+1$ degrees of freedom. How these degrees of freedom are chosen will determine the stencil.

- **ENO property**: Godunov's theorem guarantees that linear schemes of order greater than one necessarily will produce non-physical oscillations. One may circumvent this result, then, by introducing nonlinearity into the scheme. That is, the reconstruction itself depends upon a nonlinear functional of the stencil averages (2.1).

2.1.1 Smoothness Indicators

The essentially non-oscillatory, or ENO, property is introduced to limit non-physical numerical oscillations in regions with steep gradients. ENO reconstruction schemes assume that a smooth polynomial exists locally even in regions with shocks, if only we use the appropriate selection (stencil) of data to construct it. Weighted ENO, or WENO, advances this by applying nonlinear weights to combine all reconstructions into a higher-order, essentially non-oscillatory polynomial. Central to the computation of these weights is the smoothness indicating functional

$$SI_r = \sum_{l=1}^{N-1} \int_{x^{k-1/2}}^{x^{k+1/2}} \left|T^k\right|^{2l-1} \left(\frac{d^l p_r}{dx^l}\right)^2 dx, \tag{2.4}$$

where p_r is the r^{th} interpolated polynomial of degree $N-1$. SI_r is then just a sum of squares of L^2 norms of p_r (i.e., a Sobolev norm) over the cell T^k. The $\left|T^k\right|$ scaling removes mesh dependency in the functional. Assuming a uniform mesh, SI_r reduces to a simple nonlinear function of cell averages. WENO proves to be quite robust, but is unfortunately restricted to uniform mesh. We will present two different ENO-type reconstructions, [41] and [17, 18], that do not carry such a restriction. The main distinguishing factors among these and other ENO-type schemes are the smoothness

indicating functions and the weights used in combining the reconstructions.

2.2 QENO Reconstruction Scheme

In [41], an algorithm is proposed which uses data-dependent non-linear weights to construct a single polynomial. The self-described "Quasi-ENO" scheme uses a least-squares approach to reconstruct a high-order, single polynomial which preserves the average over the native element. Nonlinear, data-dependent weights are used to assign priority to nearest-neighbors' information (cell average) in the least-squares solve. A slightly modified version of this reconstruction is presented below, and is implemented in the Hybrid scheme. We will refer to this scheme as QENO.

2.2.1 Conservation of Mean

To preserve the mean, the reconstruction R_i of an N^{th}-order polynomial on cell i is

$$R_i(x - x_i) = \bar{u}_i + \sum_{k=1}^{N} \left\{ \frac{1}{k!} \frac{\partial^k u}{\partial x^k}\bigg|_i \left((x - x_i)^k - \bar{x}_i^k \right) \right\} \tag{2.5}$$

where the geometric coefficients are given as

$$\bar{x}_i^k = \frac{1}{|V_i|} \int_{V_i} (x - x_i)^k dx \tag{2.6}$$

By inspection it is seen that Eq. (2.5) preserves the mean.

2.2.2 Stencil Selection

For each cell i there is an associated stencil

$$S^i = \{i - p, i - p + 1, ..., i + q - 1, i + q\}$$

where $p \geq 0$ and $q \geq 0$ are the numbers of neighbors to include in the reconstruction on the left and and right, respectively. For implementation purposes, as will be seen later, the native cell i is not included in the stencil index S^i, as its "information" (namely, the cell average $\bar{u}_i$) is used to enforce conservation of mean and will not be included as an equation in the least-squares problem. For this research the values of $p = q = N$ are taken, thereby giving a total stencil of size $2N$. That is, there will be double the amount of information given than what is required to reconstruct a degree-N polynomial. Assuming a simple ordering of grid points ($x_{i-1} < x_i < x_{i+1}$ for all i), a volume from stencil S^i and its corresponding function average are denoted S^i_j and $\bar{u}_{i_j}$, respectively, where $1 \leq j \leq 2N$ and

$$S^i_j = i_j = \begin{Bmatrix} i + j - N - 1, & 1 \leq j \leq N \\ i + j - N & N + 1 \leq j \leq 2N \end{Bmatrix}. \tag{2.7}$$

Appropriate modifications are needed near boundaries.

2.2.3 Reconstruction Algorithm

To accomplish exact reconstruction of degree N polynomials, a least-squares procedure is used to compute the N derivative terms $\left\{ \frac{1}{k!} \frac{\partial^k u}{\partial x^k} \Big|_i \right\}_{k=1}^{N}$ in the expansion of R_i in Eq. (2.5). These derivatives will be computed by minimizing the predicted mean value of R_i over all control volumes $\left\{ \bar{u}_{i_j} \right\}_{j=1}^{2N}$ in the stencil S^i. Each of these mean values can be expressed explicitly as

$$\frac{1}{|V_{i_j}|} \int_{V_{i_j}} R_i(x - x_i) dx = \bar{u}_i + \sum_{k=1}^{N} \left\{ \frac{1}{k!} \frac{\partial^k u}{\partial x^k} \Big|_i \left(\frac{1}{|V_{i_j}|} \int_{V_{i_j}} (x - x_i)^k dx - \bar{x}_i^k \right) \right\} \tag{2.8}$$

To simplify this expression, we extract the geometric terms and define

$$\hat{x}_{ij}^k = \frac{1}{\left|V_{i_j}\right|}\int_{V_{i_j}} (x - x_i)^k dx - \bar{x}_i^k. \tag{2.9}$$

Thus, (2.8) is rewritten as

$$\frac{1}{\left|V_{i_j}\right|}\int_{V_{i_j}} R_i(x - x_i)dx = \bar{u}_i + \sum_{k=1}^{N} \frac{1}{k!}\frac{\partial^k u}{\partial x^k}\bigg|_i \hat{x}_{ij}^k \tag{2.10}$$

For neighbor V_{i_j} in the stencil of V_i, the difference in the predicted mean value and the actual value is

$$\frac{1}{\left|V_{i_j}\right|}\int_{V_{i_j}} R_i(x - x_i)dx - \bar{u}_{i_j} = \sum_{k=1}^{N}\left(\frac{1}{k!}\,\frac{\partial^k u}{\partial x^k}\bigg|_i \hat{x}_{ij}^k\right) + \left(\bar{u}_i - \bar{u}_{i_j}\right) \tag{2.11}$$

The least-squares problem is tentatively formed, then, by setting Eq. (2.11) equal to zero for all volumes in the stencil. Solving for the derivatives, this is a system of N unknowns with $2N$ equations coming from each of the volumes in the stencil.

The least-squares solution to such a system, however, enforces Eq. (2.10) equally for all neighbors in the stencil. This is clearly unacceptable, as one would like to bias the influence of neighbors in such a way that

1. preference is given to volumes physically closer to the reconstruction volume (compact support), and

2. the contribution of discontinuous data in the reconstruction is reduced to the level of truncation error (ENO property), yet

3. the contribution of smoothly connected data is unperturbed (high-order accuracy)

For reconstruction on volume V_i these items can be accomplished by appropriately

selecting weights w_{ij} that will weight each of the $j = 1, ..., 2N$ equations from (2.10) in the least squares problem.

2.2.4 Weights and the ENO Property

The weights will have two components: geometric, which can be computed during a preprocessing stage, and a smoothness indicator, which is nonlinearly data-dependent and must be computed for each reconstruction. The geometric contribution is simply

$$w_{ij}^{g} = \frac{1}{|x_j - x_i|^2} \tag{2.12}$$

The data-dependent component is more complex. It is presented as

$$w_{ij}^{DD} = 1 \Bigg/ \left(\varepsilon + \left| \frac{\bar{u}_j - \bar{u}_i}{x_j - x_i} \right|^{N+1} \right) \tag{2.13}$$

where $\varepsilon = 10^{-10}$ is chosen to prevent division by zero. The divided difference $\left| \frac{\bar{u}_j - \bar{u}_i}{x_j - x_i} \right|^{N+1}$ is the smoothness indicator.

Previous work [41] solved the least squares solution twice: once with only the geometric component of the weights (i.e. a data-independent problem), and then again with the same weights as above, except with the residual of the first results multiplied into the divided difference in the denominator of w^{DD}.

However, with the assumption that the Hybrid will only invoke the L^2-ENO scheme in regions with questionable smoothness, it is unnecessary to use the residual component as by circumstance it will automatically be significantly larger than ε and therefore scale each row of the least-squares problem equally. The weights then can be formally defined as

$$w_{ij} = w_{ij}^{g} w_{ij}^{DD} = \frac{1}{|x_i - x_{i_j}|^2} \cdot \left(\varepsilon + \left| \frac{\bar{u}_i - \bar{u}_{i_j}}{x_i - x_{i_j}} \right|^{N+1} \right)^{-1} \tag{2.14}$$

Thus, the least-squares problem on control volume i with neighbors indexed $j = 1, ..., 2N$ is the system

$$\mathbf{Ld} = \mathbf{b}, \tag{2.15}$$

where

$$\begin{aligned} L_{jk} &= w_{ij}\hat{x}_{ij}^{k} \\ d_j &= \tfrac{1}{j!}\left.\tfrac{\partial^j u}{\partial x^j}\right|_i \\ b_j &= w_{ij}(\bar{u}_{i_j} - \bar{u}_i). \end{aligned} \tag{2.16}$$

2.3 DK Reconstruction Scheme

The reconstruction algorithm presented in Sec. 2.2 constructs a single polynomial by solving a least-squares problem with different weights corresponding to the relative importance of preserving neighboring cell averages. An alternative idea follows the more traditional WENO scheme: compute multiple polynomial reconstructions over differing stencils, and then combine these together in a non-linear fashion to arrive at a final high-order, non-oscillatory reconstruction. The scheme presented is related to the procedures of [18] and [17], and is named DK after the authors of this work.

2.3.1 Polynomial Representation

One can view the fundamental degrees of freedom to be reconstructed as either nodal values or modal coefficients of the polynomial. These forms are

$$u(x) = \sum_{j=0}^{N} u(x_j) l_j(x) = \sum_{j=0}^{N} \hat{w}_j P_j(x), \tag{2.17}$$

respectively. The DK scheme reconstructs the latter.

2.3.2 Stencil Selection

The domain is tessellated into conforming elements $T^{(m)}$,

$$\Omega = \bigcup_{m=1}^{K} T^{(m)}.$$

On each of element $T^{(m)}$ a selection of n_s reconstruction stencils will be selected by considering collections of n_e neighboring elements,

$$\mathcal{S}_j^{(m)} = \bigcup_{k=1}^{n_e} T^{j(k)}, 1 \leq j \leq n_s \tag{2.18}$$

By convention we take $j(k) = m$ for all stencils, that is, the first element in each stencil is the local element itself. This is necessary for the conservation of the cell average. More precisely, the mapping function $j(k)$ carries a dependence on the local element m, and so can be thought of as $j = j^m(k)$. We will consider now an arbitrary element and so suppress this dependence on m in the notation. The problem that remains is to select $f_j(k)$.

For the one-dimensional case the stencil selection is straightforward. To reconstruct a polynomial or degree N we need at least $N + 1$ cell averages, and therefore we must have $n_e \geq N + 1$. Each stencil will have the form

$$\mathcal{S}_j^{(m)} = \{m, m - r, m - r + 1, ..., m - 1, m + 1, ..., m + s - 1, m + s\} \tag{2.19}$$

By convention the first stencil ($j = 1$) is a central stencil, chosen to contain the nearest neighbors. For this case one may take $r = s = \lfloor (N + 1)/2 \rfloor$, the greatest integer less than $(N + 1)/2$. Note that this will in some cases lead to an overdetermined system. This is acceptable, and even necessary in higher dimensions.

For the second and third stencils ($j = 2, 3$) we may pick a left and right bias,

respectively, leading to $r = N, s = -1$ and $r = -1, s = N$. Depending on whether N is odd or even, and thus whether the the first stencil is of length $N+1$ or $N+2$, it may be computationally convenient to also expand the second and third stencils to include an extra element. Thus one might instead choose $r = N+1, s = -1$ and $r = -1, s = N+1$.

Stencil selections for higher dimensions are a bit more complicated. For two-dimensional triangular elements, there are $n_s = 7$ stencils: a central stencil, three primary stencils in each of the directions normal to the sides of the element, and three additional "reverse WENO" stencils. The latter three are computed by considering instead the vertices of the triangle and then taking the elements spanned by the reverse (negative) directions of the correspodning primary stencils. In three dimensions, the number of stencils is $n_s = 9$. For a more thorough discussion of the stencil-building process, see [17].

2.3.3 Reconstruction Algorithm

Polynomial reconstructions $R^{(m)}(\mathbf{x})$ on element $T^{(m)}$ will be computed on each stencil by requiring conservation over each cell in the stencil. That is,

$$\frac{1}{|T^{(j(k))}|}\int_{T^{(j(k))}} R^{(m)}(\mathbf{x})d\mathbf{x} = \bar{u}_{j(k)}, 1 \leq k \leq n_e, j(k) \in \mathcal{S}_j^{(m)} \tag{2.20}$$

It is helpful for the sake of implementation to address how this integral will be computed. For numerical quadrature we would like to transform coordinates so that all integrals on a reference element T^E. This transformation is denoted as

$$\mathbf{x} = \mathbf{x}(T^{(m)}, \xi), \xi = \xi(T^{(m)}, \mathbf{x}) \tag{2.21}$$

The Jacobian of this transformation is

$$J_{ij} = \frac{\partial x_i}{\partial \xi_j} \tag{2.22}$$

and its determinant is $|J|$. Therefore (2.20) may be written as

$$\frac{|J|}{|T^{(j(k))}|}\int_{\tilde{T}^{(j(k))}} R^{(m)}(\xi)d\xi = \bar{u}_{j(k)}, 1 \leq k \leq n_e, j(k) \in \mathcal{S}_j^{(m)} \tag{2.23}$$

The integrals in Eq. (2.23) are over the $(j(k))$-indexed elements in the ξ-coordinate space; still, for the sake of implementation, these will be each transformed *again*, this time to the reference element T^E in $\tilde{\xi}$-coordinate space. This transformation is denoted as

$$\xi = \xi(\tilde{T}^{(j(k))}, \tilde{\xi}), \tilde{\xi} = \tilde{\xi}(\tilde{T}^{(j(k))}, \xi) \tag{2.24}$$

Similar to Eq. (2.22), the Jacobian of this second transformation is denoted $\tilde{J}$. Now, the element to be integrated over is simply the reference element T^E and so the conservation equations (2.23) can be written as

$$\frac{|J||\tilde{J}|}{|T^{(j(k))}|}\int_{T^E} R^{(m)}(\xi(\tilde{T}^{(j(k))}, \tilde{\xi}))d\tilde{\xi} = \bar{u}_{j(k)}, 1 \leq k \leq n_e, j(k) \in \mathcal{S}_j^{(m)} \tag{2.25}$$

Assuming the case of straight-edged elements (e.g. triangles and tetrahedrons), the Jacobian J is constant and we have the relationship

$$|J| = \frac{|T^{(j(k))}|}{|\tilde{T}^{(j(k))}|}. \tag{2.26}$$

Likewise, we have

$$|\tilde{J}| = \frac{|\tilde{T}^{(j(k))}|}{|T^E|} \tag{2.27}$$

Thus, the Jacobian factors cancel out and, using the convention of summation over repeated indices and recalling the representation from Eq. (2.17), the conservations equations may be written

$$\left(\frac{1}{|T^E|}\int_{T^E} \Psi_l(\xi(\tilde{T}^{(j(k))}, \tilde{\xi}))d\tilde{\xi}\right)\hat{w}_l = \bar{u}_{j(k)}, 1 \leq k \leq n_e, j(k) \in \mathcal{S}_j^{(m)} \tag{2.28}$$

Using the convention of summation over repeated indices, Eq. (2.28) can be succinctly written in tensor notation as

$$A_{kl}\hat{w}_l = \bar{u}^k \tag{2.29}$$

where

$$A_{kl} = \left(\frac{1}{|T^E|} \int_{T^E} \Psi_l(\xi(\tilde{T}^{(j(k))}, \tilde{\xi}))d\tilde{\xi} \right) \hat{w}_l \tag{2.30}$$

and

$$\bar{u}^k = \bar{u}_{j(k)} \tag{2.31}$$

We finally have a linear system to solve which will produce a reconstruction for stencil j, $1 \leq j \leq n_s$. If we denote the number of degrees of freedom to be solved for as N_p ($N+1$ in one dimension and $\frac{1}{2}(N+1)(N+2)$ in two dimensions), the number of neighbors n_e (including the native element itself) needed for reconstruction must be greater than or equal to N_p. In one dimension we may take $n_e = N_p$ or $n_e = N_p + 1$, depending on whether one would like a balanced, central stencil (that is, equal number of left- and right- neighbors). However for higher dimensions a greater stencil size should be included, so as to handle any potential singularities in the linear system that may arise due to the unstructured mesh. In [17] the values $n_e = 1.5N_p$ and $n_e = 2N_p$ are suggested for two and three dimensions, respectively.

Ensuring Conservation of Mean

The linear system (2.29) will become overdetermined when $n_e > N_p$, and so an additional constraint must be added to ensure each reconstruction is mean-preserving over its native element $T^{(m)}$. Recall that the function representation is $\Psi_l \hat{w}_l$, where Ψ_l are the basis functions and $\hat{w}_l$ are the degrees of freedom. If we use Legendre polynomials as basis functions, the necessary constraint is

$$\hat{w_0} = \frac{1}{\gamma_0}\bar{u}^{j(1)} = \frac{1}{\gamma_0}\bar{u}^{(m)} \tag{2.32}$$

where γ_0 is the normalization constant, dependent upon whether one is using normalized ($\gamma_0 = \sqrt{2}$) or non-normalized ($\gamma_0 = 1$) basis functions.

This constraint is written in tensor notation as

$$C_l \hat{w}_l = \frac{1}{\gamma_0} R_i \bar{u}_i \tag{2.33}$$

where all entries of C_l and R_i are zero except for their first, $C_0 = \frac{1}{\gamma_0}$, and $R_1 = 1$. Note that C_l is indexed as $0 \leq l \leq N$ and R_i is indexed as $1 \leq i \leq n_s$. The constraint (2.32) is obtained, then, by the minimizer of the functional

$$f = \left(A_{kl}\hat{w}_l - \bar{u}^k\right) \cdot \left(A_{kj}\hat{w}_j - \bar{u}^k\right) - \lambda \cdot \left(C_l \hat{w}_l - R_i \bar{u}_i\right) \tag{2.34}$$

over the degrees of freedom $\hat{w_l}$ with Lagrange multiplier λ. Differentiating with respect to these, we get the equations

$$\frac{\partial f}{\partial \hat{w}_l} = 2A_{kl}A_{kj}\hat{w}_j - 2A_{kl}\bar{u}^k - \lambda C_l = 0, \frac{\partial f}{\partial \lambda} = C_l \hat{w}_l - R_i \bar{u}_i = 0 \tag{2.35}$$

The final linear system to solve, then, that will incorporate these constraints and conserve the mean, is

$$\begin{pmatrix} 2A_{kl}A_{kj} & -C_l \\ C_l \delta_{lj} & 0 \end{pmatrix} \cdot \begin{pmatrix} \hat{w}_l \\ \lambda \end{pmatrix} = \begin{pmatrix} 2A_{kl}\bar{u}^k \\ R_i \bar{u}_i \end{pmatrix} \tag{2.36}$$

where δ_{lj} is the Kronecker delta which returns one if the two integers l, j are equal, and zero otherwise. This system completely reconstructs the polynomial degrees of freedom while conserving the mean. For clarity, it is noted that the top left block $2A_{kl}A_{kj}$ is size N_p-by-N_p, the bottom left $C_l\delta_{lj}$ is size 1-by-N_p, the top right $-C_l$ is size N_p-by-1, and the bottom right 0 is a singleton.

For the sake of implementation, the left hand side matrix in equation 2.36 will be inverted and stored for each stencil on each element. This is done as a preprocessing stage, once and for all. Then, each reconstruction may be computed by forming the cell average-dependent right hand side and multiplying by the stored matrix.

2.3.4 Weights and the ENO Property

Having computed each of the n_s reconstructions, the next step is to weigh them in such a way as to preserve high-order accuracy while eliminating non-physical oscillations. The final reconstruction, $R_{WENO}(\xi)$, then will be

$$R_{WENO}(\xi) = \sum_{i=1}^{n_s} \omega_i R_i(\xi) = \left(\sum_{i=1}^{n_s} \omega_i \hat{w}_l^i \right) \Psi_l(\xi) \tag{2.37}$$

where ω_i are nonlinear weights which will be determined by considering the smoothness of the polynomial reconstructions.

To maintain the preservation of the cell average, it is necessary that $\sum_{i=1}^{n_s} = 1$, so we the have normalization relationship

$$\omega_i = \frac{\tilde{\omega}_i}{\sum_{j=1}^{n_s} \tilde{\omega}_j} \tag{2.38}$$

The (non-normalized) weights $\tilde{\omega}_i$ are of the form

$$\tilde{\omega}_i = \frac{\lambda_i}{(\epsilon + \sigma_i)^r}. \tag{2.39}$$

ϵ is a small number close to zero (e.g. 10^{-8}), chosen to avoid division by zero. The exponent r is an added nonlinearity to the weights, given to magnify the contribution of smooth reconstructions to the stencil and further penalize reconstructions with oscillations detected. The value $r = 4$ has been chosen in this work, per suggestion from the original authors.

A further, less obvious aspect of the scheme is the stencil biasing parameter λ_i.

This is given as

$$\lambda_i = \begin{cases} \lambda_1, & i = 1 \\ 1, & \text{else} \end{cases} . \tag{2.40}$$

The value of λ_1 may be adjusted to give an artificial preference to the central stencil, which aids in maintaining high-order accuracy of the reconstruction at the possible risk of allowing oscillations. For the purpose of robustness it would be preferable to have a definite, set value of λ_1, but in practice one finds that for best results it is problem-dependent. An acceptable value is on the order of 10^3, while for problems with high-order features (for example, the shock density wave equation in Sec. 3.2.4) that need to be accurately resolved, it may be preferable to increase this to be more in the range of 10^6. Conversely, for problems with mostly piecewise linear data such as Sod's shock tube problem (Sec. 3.2.4) it may be preferable to take a lower value for λ_1, perhaps even as low as 1 (no biasing of the central stencil).

In the implemention of the standard WENO scheme [51], the explicit, reconstructed quantities are function values at the cell boundaries. As a result, the smoothness indicators simplify to a non-linear combination of cell averages. Complications arise, then, when considering problems with irregular or unstructured mesh. In the DK reconstruction, however, the entire set of polynomial degrees of freedom $\tilde{w}_i$ are reconstructed, and so the smoothness indicator may be computed as a quadratic functional of these degrees of freedom, seen in equation (2.41).

$$\sigma_i = \sum_{|\alpha|=1}^{N} \int_{T^E} \left(\frac{\partial^{|\alpha|}}{\partial \xi^{\alpha}} w_i(\xi) \right)^2 d\xi \tag{2.41}$$

Here, $\alpha = (\alpha_1, \ldots, \alpha_n), |\alpha| = \alpha_1 + \ldots + \alpha_n$ is a multi-index. Note that because we are working in the reference system, the mesh scaling has been removed from the equation. Eq. (2.41) can be succinctly written as

$$\sigma_i = \hat{w}_l^i \Sigma_{lm} \hat{w}_m^i, l, m = 1, 2, \ldots, N \tag{2.42}$$

where the matrix Σ_{lm} acts on the degrees of freedom corresponding to the higher order (non-constant) basis functions and is defined as

$$\Sigma_{lm} = \sum_{|\alpha|=1}^{N} \int_{T^E} \frac{\partial^{|\alpha|}}{\partial \xi^{\alpha}} \Psi_l(\xi) \cdot \frac{\partial^{|\alpha|}}{\partial \xi^{\alpha}} \Psi_m(\xi) d\xi. \tag{2.43}$$

As all computations are performed over the reference element T^E, the matrix Σ_{lm} does not depend on the mesh and is therefore a *universal* smoothness indicator. It may be computed once and for all, using Gaussian quadrature to evaluate the integrals.

Thus far, the reconstruction has been presented as for a scalar function $u = \sum_j \hat{w}_j P_j(x)$. A discussion in [18], however, details additional features of the scheme when we consider systems of conservation laws. In particular, weighted reconstructions are performed on characteristic variables, in addition to conserved ones *in each normal direction* to the element, from among which the reconstruction of each component is chosen. For the results of the one-dimensional Euler system of equations presented in Sec. 3.2.4, this implementation is adopted.

Chapter 3

The Hybrid Scheme

3.1 Switching Between Schemes

The discontinuous Galerkin scheme offers high-order accuracy on an arbitrary mesh, provided that the underlying solution is sufficiently smooth. Furthermore, the computational cost of implementing DG is generally much cheaper than that of a finite volume scheme of similar accuracy, especially if one chooses to use non-linear ENO function reconstructions for the numerical flux. It is desired, then, to maintain the DG scheme as much as possible and to switch to the high-order ENO scheme only when necessary.

One such way of accomplishing this is via a limiter. Rather than actually changing the solution, however, we seek to use the limiter as simply a means to judge the smoothness of the solution. The generalized slope limiter presented in [9] provides such a framework.

We outline a shock-detecting procedure by first defining the minmod function

$$m(a_1, ..., a_n) = \begin{cases} s \min_{1 \le i \le n} |a_i|, & |s| = 1 \\ 0, & \text{otherwise,} \end{cases} \qquad s = \frac{1}{n} \sum_{i=1}^{n} \text{sign}(a_i) \tag{3.1}$$

The minmod function will return 0 if the signs of the input a_i are not in agreement,

and otherwise will return the smallest (in magnitude) of the a_i. From this, we have the early workings of a possible switch parameter for the Hybrid scheme: given cell averages in a suitably local stencil as inputs to m, we could opt to switch schemes based on the output. The following approach is taken for one dimensional problems:

Define left and right interface flux approximations as

$$\begin{aligned} v_l^k &= \bar{u}_h^k - m\left(\bar{u}_h^k - u_l^k, \bar{u}_h^k - \bar{u}_h^{k-1}, \bar{u}_h^{k+1} - \bar{u}_h^k\right), \\ v_r^k &= \bar{u}_h^k + m\left(u_r^k - \bar{u}_h^k, \bar{u}_h^k - \bar{u}_h^{k-1}, \bar{u}_h^{k+1} - \bar{u}_h^k\right). \end{aligned} \tag{3.2}$$

where u_l^k and u_r^k refer to the solution at the first (left-most) and last (right-most) gridpoints of the element, respectively. We then make the following judgment over whether a given volume is considered a "DG element" or an "FV element":

$$\begin{aligned} \textbf{If} \quad & \left|v_l^k - u_l^k\right| < \epsilon_0 \\ \textbf{and} \quad & \left|v_r^k - u_r^k\right| < \epsilon_0, \\ \textbf{then} \quad & \text{element } T^k \text{ is tentatively considered a DG element.} \\ \textbf{Else}, \quad & \text{element } T^k \text{ is an FV element.} \end{aligned} \tag{3.3}$$

Additionally, once all FV elements have been decided upon, all nearest-neighbors are included as well. While such a choice increases the cost and potentially diminishes the accuracy by decreasing the number of DG elements, numerical observations indicate that it is necessary to include the nearest neighbors to prevent spurious oscillations.

From (3.3), then, there arises a single parameter ϵ_0 which regulates which scheme is to be used. We will refer to this number as the *switching parameter*. Generally, it is solution- and equation-dependent, although for the test problems values in the range of $10^{-2} - 10^{-4}$ are sufficient.

The method proposed to detect shock regions of the solution is only detailed for one-dimensional problems due to the nature of the limiter involved. For higher dimensions the approach would need to be modified; this is not discussed, as in practice the more difficult task is to find a suitable reconstruction algorithm that

can provide high-order, monotonic data on arbitrary mesh. Therefore, any sort of detection approach that errs on the side of over-inclusion of FV regions should initially suffice.

3.1.1 Implementation Algorithm

The Hybrid scheme is adapted from the LSERK procedure given in Eq. (1.14) for a fourth-order in time, explicit Runge-Kutta time scheme. We assume that at the outset of the time integration, the solution $u_h^k(x)$ is known for all elements T^k everywhere, regardless of whether T^k is in a smooth or non-smooth region. We then simultaneously evolve

$$\frac{du_h^{DG}}{dt} = L_h(u_h, t) \tag{3.4}$$

and

$$\frac{d\bar{u}_h^{FV}}{dt} = \bar{L}_h(\bar{u}_h, t). \tag{3.5}$$

where u_h^{DG} is a nodal representation of the solution in the smooth regions, and $\bar{u}_h^{FV}$ are the cell averages of the solution in non-smooth regions.

Adapting the same conventions as in (1.14) where additional storage variables are labeled v and inter-stage solutions are labeled p, the scheme is as follows:

1. Before beginning the inner Runge-Kutta loops, compute which elements are in declared shock regions using (3.3)

2. Begin inner Runge-Kutta loops: having all necessary solution data available, compute Runge-Kutta residuals v^{DG} and $\bar{v}^{FV}$, and the updated solutions p^{DG} and $\bar{p}^{FV}$

3. Extract cell averages from smooth region (DG) elements

4. Having all necessary cell averages available, compute reconstructions in all shock region (FV) elements

5. Repeat (2)-(4) for each of the 5 LSERK stages

6. Return to (1) for the next time step

The algorithm will minimize the number of reconstructions needed, as this tends to be the greatest cost. For CFL time stepping restrictions, one may also need to consider the eigenvalues of both the L_h and $\bar{L}_h$ operators.

3.2 Results

In this section we will examine numerical simulations involving the Hybrid scheme. For a variety of transient simulations we will answer the following questions:

- Are we able to prevent non-physical oscillations?
- Are we able to accurately capture the discontinuity?
- What parameter sensitivities do we observe?
- How are the results affected by mesh refinement?

3.2.1 Convergence Studies

A fundamental property of the reconstruction algorithms is the ability to fully represent smooth functions up to the order of the discretization. To test the $O(h^{N+1})$ convergence properties the reconstructions are applied to the function

$$f(x) = \exp\left[\sin\left(x\right)\right], x \in \left[0, 2\pi\right]. \tag{3.6}$$

Convergence studies are performed for each reconstruction and presented in Tables 3.1 and 3.2. These confirm $O(h^{N+1})$ accuracy.

		Q-ENO		DK	
	K	L_∞-error	Convergence rate	L_∞-error	Convergence rate
$N=2$	20	1.93e-02		3.07e-02	
	40	2.22e-03	3.12	4.29e-03	2.84
	80	2.72e-04	3.03	5.51e-04	2.96
	160	3.35e-05	3.02	6.93e-05	2.99
$N=3$	20	6.12e-03		7.58e-03	
	40	8.15e-04	2.91	4.46e-04	4.09
	80	6.00e-05	3.76	2.50e-05	4.16
	160	3.88e-06	3.95	1.50e-06	4.06
$N=4$	20	7.89e-03		1.51e-02	
	40	2.48e-04	4.99	6.80e-04	4.47
	80	5.79e-06	5.41	2.38e-05	4.83
	160	1.91e-07	4.92	7.65e-07	4.86

Table 3.1: Convergence study for DK and Q-ENO reconstructions. L_∞ error is given for K elements of order N.

		Q-ENO		DK	
	K	L_1-error	Convergence rate	L_1-error	Convergence rate
$N=2$	20	1.13e-02		2.58e-02	
	40	1.42e-03	2.99	3.22e-03	3.00
	80	1.85e-04	2.95	4.00e-04	3.01
	160	2.35e-05	2.97	4.98e-05	3.01
$N=3$	20	6.61e-03		7.60e-03	
	40	4.32e-04	3.93	3.27e-04	4.53
	80	2.44e-05	4.15	1.58e-05	4.37
	160	1.28e-06	4.25	8.90e-07	4.15
$N=4$	20	4.78e-03		1.67e-02	
	40	1.19e-04	5.33	6.33e-04	4.72
	80	4.20e-06	4.82	2.12e-05	4.90
	160	1.51e-07	4.80	6.73e-07	4.98

Table 3.2: Convergence study for DK and Q-ENO reconstructions. L_1 error is given for K elements of order N. For DK we use a biasing parameter of $\lambda_1 = 10^3$.

3.2.2 One-dimensional Advection Equation

The one-dimensional advection equation is given as

$$\begin{aligned} \tfrac{\partial u}{\partial t} + c\tfrac{\partial u}{\partial x} = 0, x \in [a,b] \\ u(x,0) = u_0(x) \end{aligned} \tag{3.7}$$

The initial condition u_0 is assumed to be periodically extended across $[a,b]$, and so the analytic solution to this equation is $u(x,t) = u_0(x-ct)$. That is, the solution is a traveling wave with speed c. As the solution of the PDE is simply a propagation of the initial condition $u_0(x)$, the smoothness of the solution is completely determined by the initial condition as well.

Riemann Problem

For our first example, consider the Riemann problem

$$\begin{aligned} \tfrac{\partial u}{\partial t} + 2\pi\tfrac{\partial u}{\partial x} = 0, x \in [0,2\pi] \\ u(x,0) = \begin{cases} 0, 0 \leq x < \pi \\ 1, \pi \leq x \leq 2\pi \end{cases} \end{aligned} \tag{3.8}$$

In Figure 3.1 we present results for varying polynomial order and number of elements. The switching criteria $\epsilon_0 = 10^{-4}$ is used; there is little sensitivity to this value. It is observed that h-refinement, corresponding to higher number of elements K, appears to be more crucial to resolving the discontinuity. This is not surprising given that the solution in smooth regions is piecewise constant.

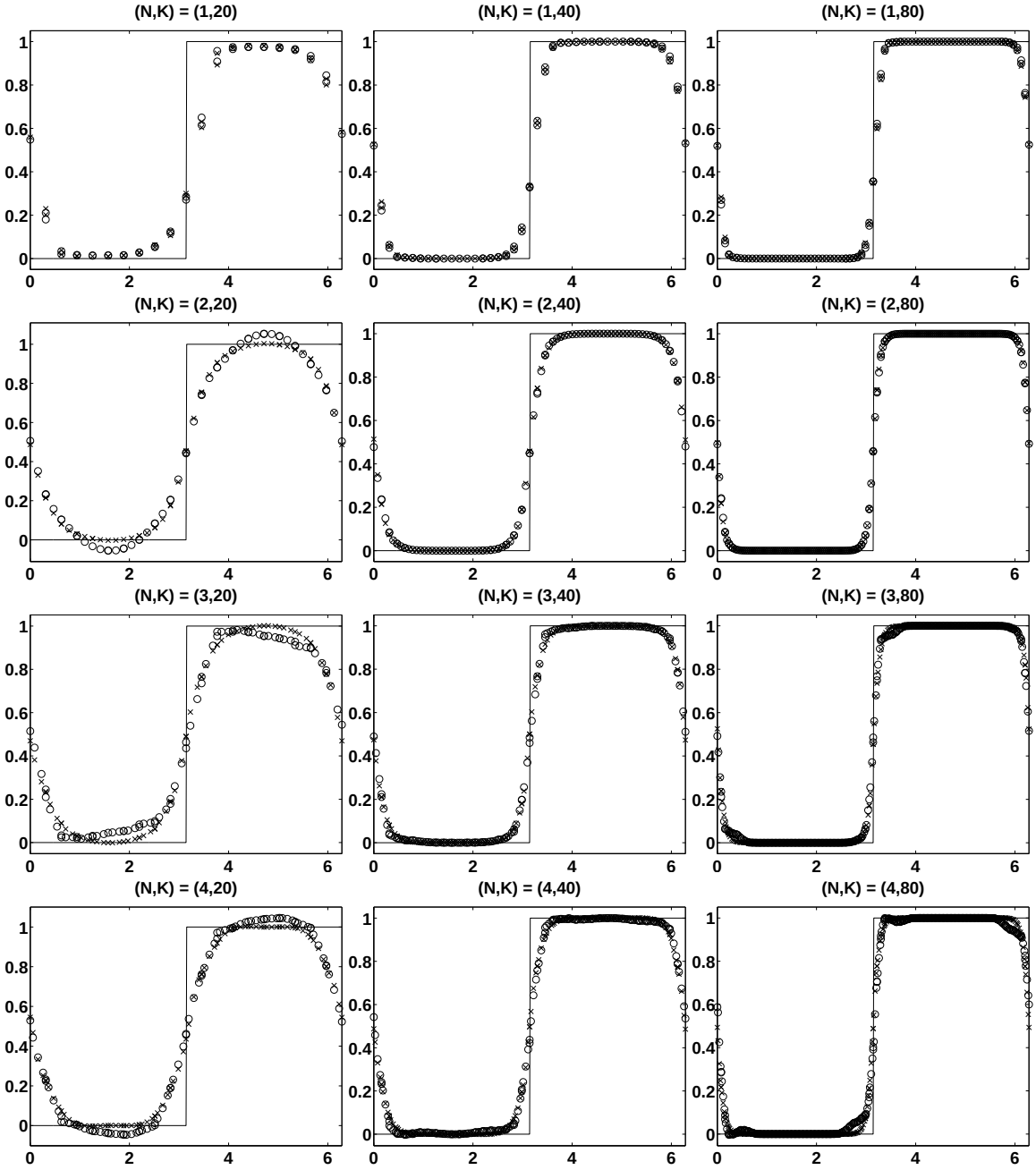

Figure 3.1: One-dimensional advection, Riemann problem (see Eq. (3.8)). Terminal time is $t = 1$. O's: Q-ENO reconstruction. X's: DK reconstruction. Solid line: exact solution. Left to right: $K = 20, 40, 80$. Top to bottom: $N = 1, 2, 3, 4$. The order of polynomial reconstruction is the same as the order of the DG scheme. A switch tolerance of $\epsilon_0 = 10^{-4}$ is used.

Advection With Mixed Features

Next we consider (3.7) with wave speed $c = 1$, domain $x \in [-1, 1]$ and initial condition

$$u(x,0) = \begin{cases} \frac{1}{6}[G(x,\beta,z-\delta) + 4G(x,\beta,z) + G(x,\beta,z+\delta)], & x \in [-0.8,-0.6] \\ 1, & x \in [-0.4,-0.2] \\ 1 - |10(x-0.1)|, & x \in [0,0.2] \\ \frac{1}{6}[F(x,\alpha,a-\delta) + 4F(x,\alpha,a) + F(x,\alpha,a+\delta)], & x \in [0.4,0.6] \\ 0, & \text{else} \end{cases} \tag{3.9}$$

where

$$\begin{aligned} F(x,\alpha,a) &= \sqrt{\max(1-\alpha^2(x-a)^2, 0)} \\ G(x,\beta,z) &= \exp(-\beta(x-z)^2) \end{aligned} \tag{3.10}$$

and $z = -0.7, \delta = 0.005, \beta = \frac{\log 2}{36\delta^2}, a = 0.5$, and $\alpha = 10$.

The initial condition (3.9), given by Jiang and Shu [30], consists of a Gaussian, a square wave, a triangle and a semi-ellipse. This particular example is very instructive in measuring the Hybrid's ability both to capture discontinuous data and to accurately resolve smooth features with steep gradients.

We perform tests first on a mesh fixed with $K = 200$ elements, but with varying order of $N = 1, 2, 3$ and 4. A switch tolerance of $\epsilon_0 = 10^{-4}$ is used. Results given in Figure 3.2 indicate that for the low-order $N = 1$ solution there is excessive smearing of all features. For sufficiently small number of elements K this tends to be the result as the features of the problem are under-resolved.

Increasing the polynomial order of approximation, we observe that the solution is well-captured for $N = 2$ and $N = 3$. However, spurious oscillations emerge $N = 4$. This is attributed to an over-ambitious search within the polynomial reconstructions where the underlying data is unsuitable for such a high-order reconstruction. One way to resolve this is to vary the switching parameter $\epsilon_0 = 10^{-2}$. In Figure 3.3 we display results where a looser value, $\epsilon_0 = 10^{-2}$, has been chosen. This in effect allows

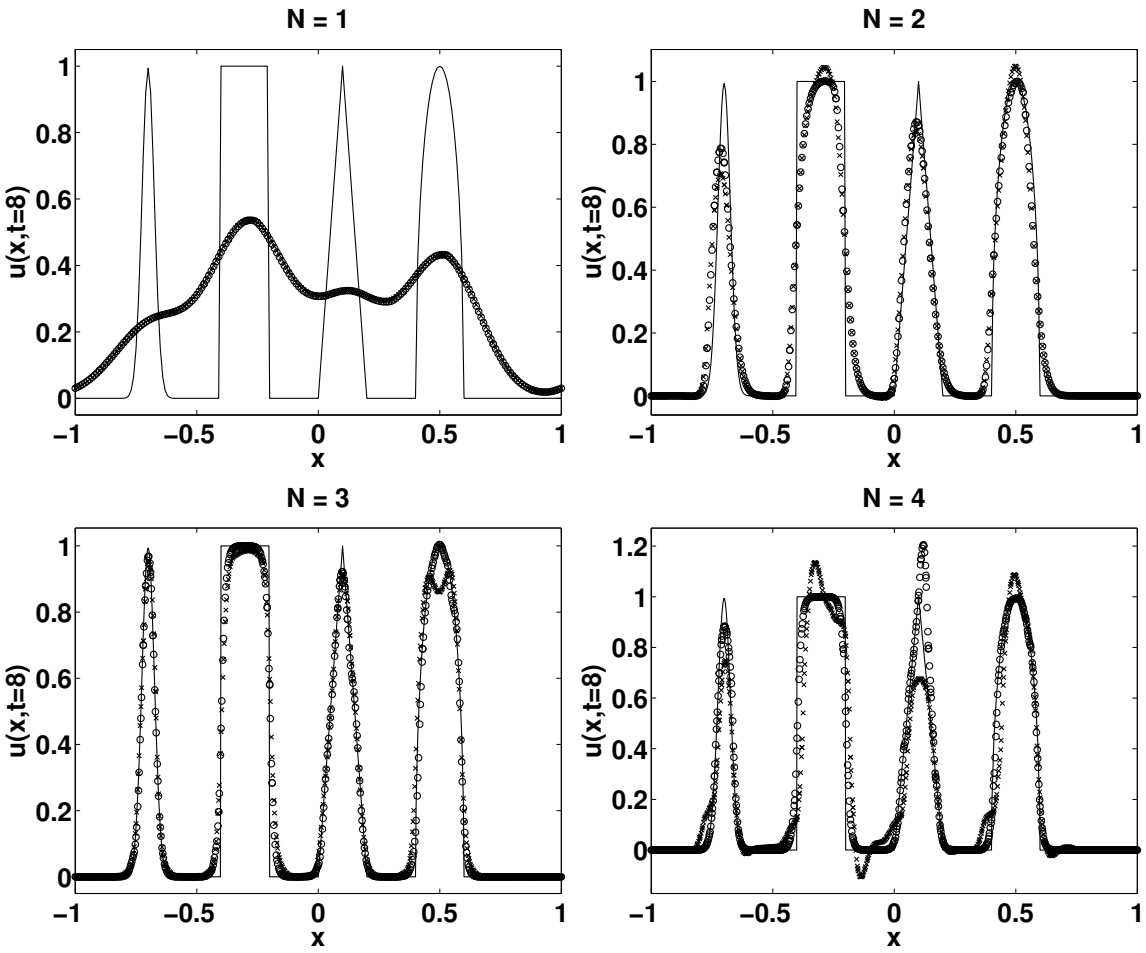

Figure 3.2: Hybrid solution for one-dimensional advection equation with initial condition given by Eq. (3.9). Q-ENO reconstruction is given by O's; DK is given by X's. Exact solution is given by solid line. $K = 200$ elements are used, with varying orders $N = 1, 2, 3, 4$. Switch tolerance is fixed at $\epsilon_0 = 10^{-4}$.

the discontinuous Galerkin method to come back in effect in shock regions, if the shocks have been sufficiently dissipated. While not a perfect solution, it does help to prevent a scenario where inaccurate reconstructions manage to aggravate numerical oscillations, rather than to limit them.

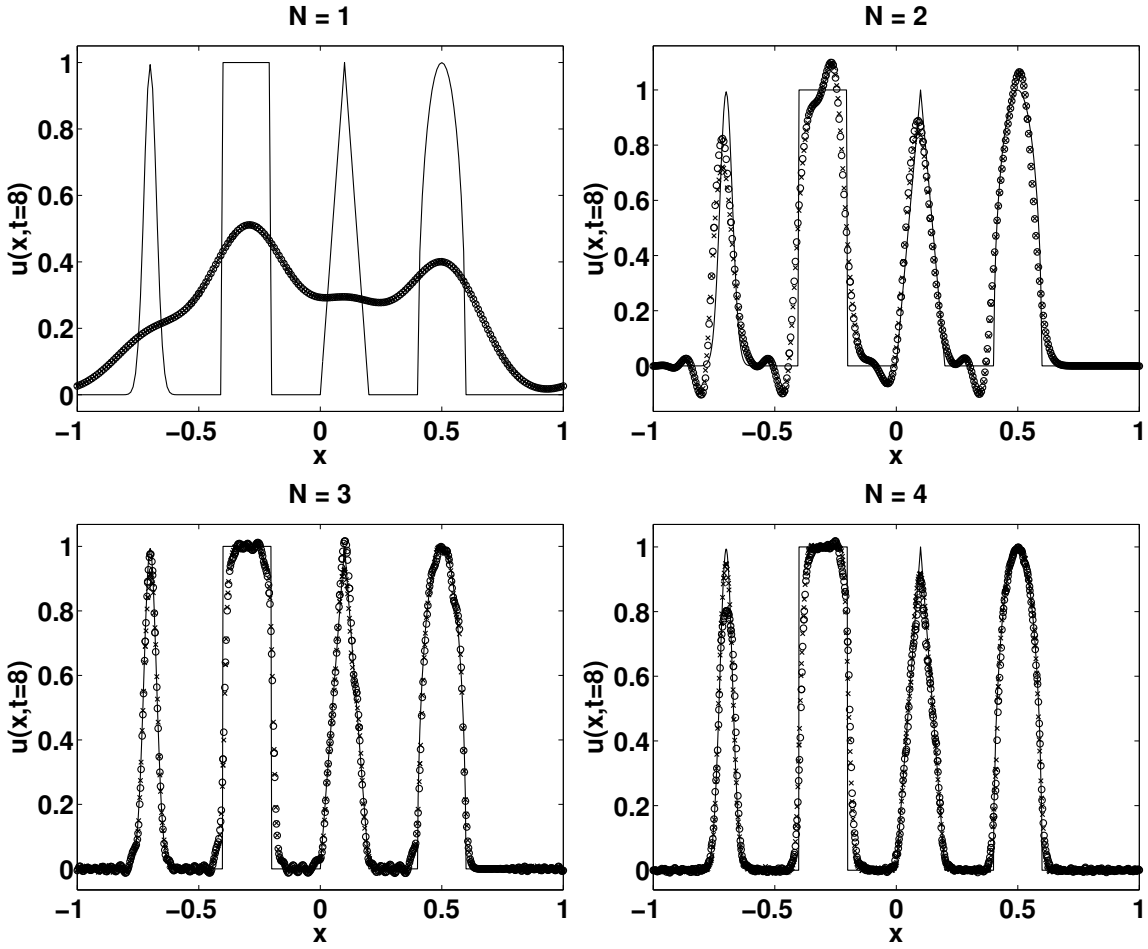

Figure 3.3: Hybrid solution for one-dimensional advection equation with initial condition given by Eq. (3.9). Q-ENO reconstruction is given by O's; DK is given by X's. Exact solution is given by solid line. A mesh of $K = 200$ elements are used, with varying order $N = 1, 2, 3, 4$. Switch tolerance is $\epsilon_0 = 10^{-2}$

3.2.3 Burgers' Equation

The one-dimensional Burgers' equation is given as

$$\begin{gathered} \frac{\partial u}{\partial t} + \left(\frac{u^2}{2}\right)_x = 0, x \in [-1, 1] \\ u(x, 0) = u_0(x) \end{gathered} \tag{3.11}$$

Due to the nonlinearity u^2 in the flux term of Eq. (3.11), the solution may develop a shock in finite time, even with a smooth initial condition $u_0(x)$.

The Hybrid scheme is applied to the one-dimensional Burgers' Equation with the initial condition

$$u_0(x) = -\sin(\pi x) \tag{3.12}$$

The exact solution to (3.11) with initial condition (3.12) is given by the Lax-Oleinik formula [20]:

$$u(x, t) = \frac{x - y(x, t)}{t} \tag{3.13}$$

where for each (x, t) pair, $y(x, t)$ can be most easily computed by solving the for the real zeros y of the function

$$-\frac{(x - y)}{t} - \sin(\pi y) = 0. \tag{3.14}$$

Convergence studies are performed by examining the solution at time $t = 0.2$, which occurs before the shock has formed. For these simulations we use the Hybrid and fix the "shock" regions to be every element in the domain; equivalently, we simply have a finite volume scheme with Q-ENO and DK reconstructions to compute numerical fluxes. These results are presented in Figure 3.3 and 3.4.

To better understand the features of the problem, exact and numerical solutions are displayed in Figure 3.4. Error plots for these solutions are given in Figure 3.5.

If we were to animate the solution of (3.11)-(3.12), we would observe the positive

		Q-ENO		DK	
	K	L_∞-error	Convergence rate	L_∞-error	Convergence rate
$N=1$	20	2.07e-01		1.75e-01	
	40	4.73e-02	2.13	2.97e-02	2.56
	80	9.15e-03	2.37	9.81e-03	1.60
	160	3.51e-03	1.38	4.15e-03	1.24
$N=2$	20	9.70e-02		1.21e-01	
	40	2.01e-03	2.27	3.49e-02	1.79
	80	3.52e-04	2.51	7.04e-03	2.31
	160	4.84e-04	2.86	1.04e-03	2.75
$N=3$	20	1.35e-01		6.82e-02	
	40	2.10e-02	2.68	1.45e-02	2.23
	80	1.47e-03	3.84	1.48e-03	3.29
	160	8.51e-06	4.11	7.85e-05	4.24
$N=4$	20	9.99e-02		8.84e-02	
	40	1.45e-02	2.78	2.19e-02	4.72
	80	9.95e-04	3.86	2.78e-03	4.90
	160	4.70e-05	4.41	1.72e-04	4.98

Table 3.3: Convergence study of Burgers' equation with smooth initial condition $u(x,0) = -\sin(\pi x)$. L_∞ errors are computed at time $t = 0.2$. Results are given using a A pure finite volume scheme with Q-ENO and DK reconstructions

		Q-ENO		DK	
	K	L_1-error	Convergence rate	L_1-error	Convergence rate
$N=1$	20	6.79e-02		7.08e-02	
	40	1.67e-02	2.03	1.59e-02	2.16
	80	3.61e-03	2.21	3.51e-03	2.18
	160	8.67e-04	2.06	8.14e-04	2.11
$N=2$	20	1.19e-02		1.79e-02	
	40	1.66e-03	2.85	3.05e-03	2.55
	80	2.81e-04	2.56	4.89e-04	2.64
	160	3.80e-05	2.88	6.56e-05	2.90
$N=3$	20	1.38e-02		1.12e-02	
	40	1.22e-03	3.50	1.57e-03	2.84
	80	8.65e-05	3.82	7.81e-05	4.33
	160	4.39e-06	4.30	3.40e-06	4.52
$N=4$	20	9.32e-03		1.91e-02	
	40	1.01e-03	3.21	2.78e-03	2.78
	80	3.74e-05	4.76	1.85e-04	3.92
	160	1.76e-06	4.40	8.19e-06	4.49

Table 3.4: Convergence study of Burgers' equation with smooth initial condition $u(x,0) = -\sin(\pi x)$. L_1 errors are computed at time $t = 0.2$. A pure finite volume scheme with Q-ENO and DK reconstructions is used

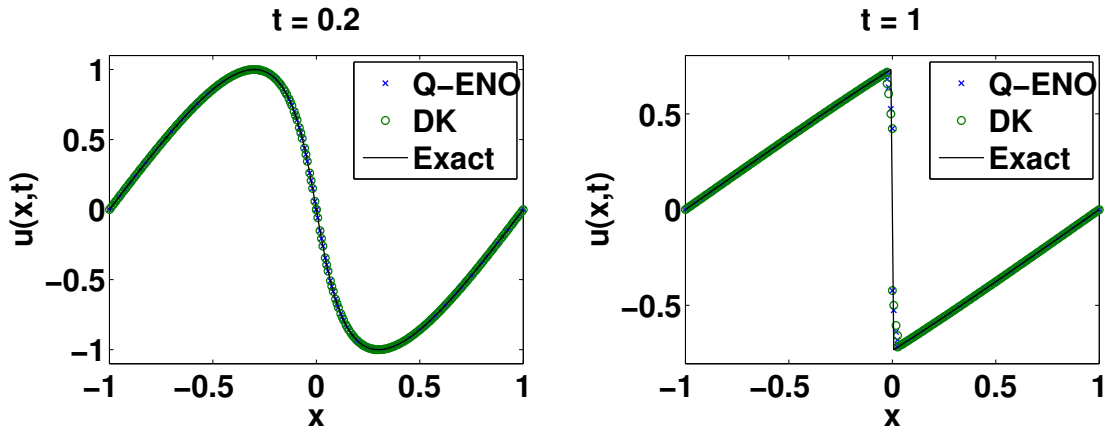

Figure 3.4: Solution to Burgers' equation (3.11)-(3.12) at times $t = 0.2$ (left) and $t = 1$ (right). A uniform mesh of order $N = 3$ and $K = 80$ elements is used.

(left) side of the initial sine wave traveling rightwards, and the negative (right) side traveling leftwards. This forms the shock at $x = 0$ that is observed in the right graph of Figure 3.4. Naturally, then, the error (in both pre- and post-shock times) is greatest in this central region, as is seen in Figure 3.5.

Recall, however, that the principle feature of the Hybrid is to operate on unstructured mesh. The Burgers' equation (3.11)-(3.12) provides an ideal test problem for this feature, as we know a priori that there will be a shock formed at $x = 0$. Therefore, we may fix a mesh that refines cell widths around this region. To this end, we prescribe the mapping

$$V_k \rightarrow \text{sign}(V) \cdot V_k^{1.5}, k = 1, 2, ..., K + 1 \tag{3.15}$$

where V_k correspond to the vertices of the K elements. Therefore, as we approach the shock region, elements become smaller, providing greater resolution of the solution.

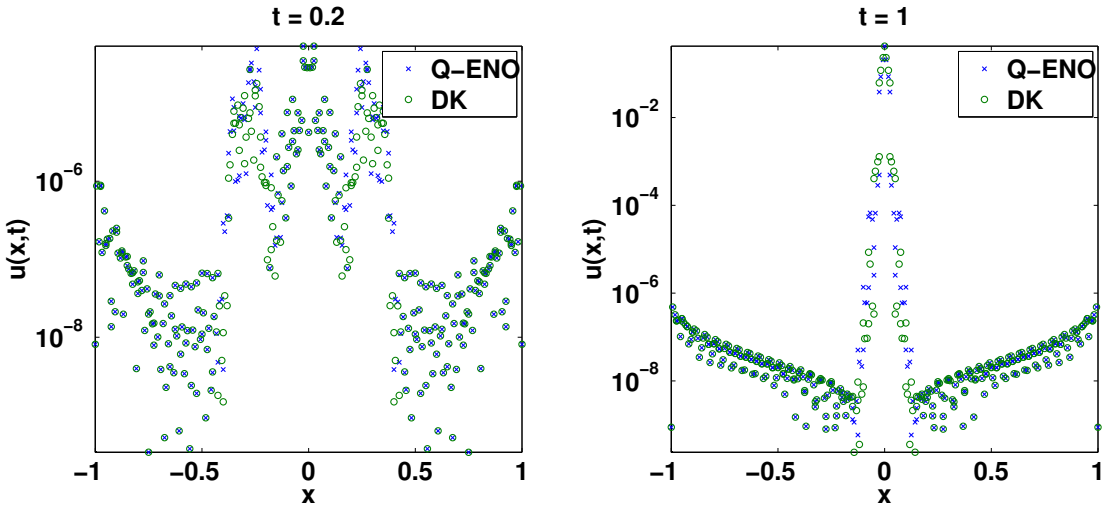

Figure 3.5: Error plots corresponding to the simulations of Burgers' equation in Figure 3.4 are given.

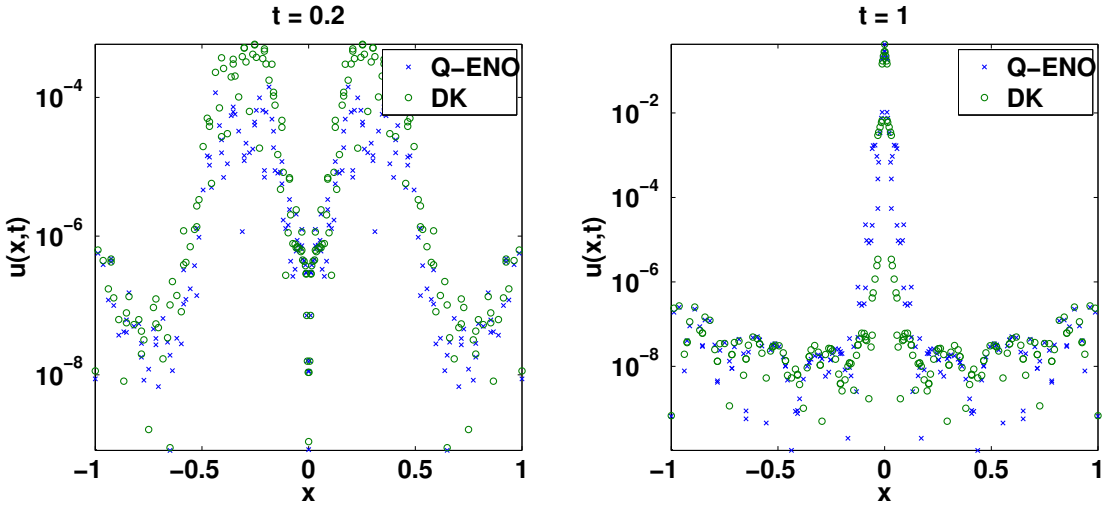

Figure 3.6: Solution to Burgers' equation (3.11)-(3.12) at times $t = 0.2$ (left) and $t = 1$ (right). A non-uniform mesh of order $N = 4$ and $K = 40$ elements is used, with element stretching corresponding to Eq. (3.15).

3.2.4 One-dimensional Euler Equations

The Hybrid scheme is tested on the 1-D system of incompressible Euler equations, given by

$$\begin{gathered} \mathbf{Q}_t + \mathbf{F}_x = 0 \\ \mathbf{Q} = (\rho, \rho u, E)^T, \mathbf{F} = (\rho u, \rho u^2 + P, (E+P)u)^T, \end{gathered} \tag{3.16}$$

with the equation of state given by

$$P = (\gamma - 1)(E - \frac{1}{2}\rho u^2), \tag{3.17}$$

and $\gamma = 1.4$ is taken.

For the Q-ENO reconstruction, the density ρ is used to compute the nonlinear smoothness indicator function; see (2.14). Moreover, this reconstruction is applied to the conserved variables. However, the DK method performs reconstruction on characteristic variables. Unless otherwise stated, for all tests we use a switching parameter value of $\epsilon_0 = 10^{-4}$.

Sod's Problem

Sod's shock tube problem presents a simple test of the Hybrid for the one-dimensional Euler Equations. The initial condition is the Riemann problem

$$(\rho, u, P)(x, 0) = \begin{cases} (1, 0, 1), & x \in [0, 0.5) \\ (0.125, 0, 0.1), & x \in [0.5, 1]. \end{cases} \tag{3.18}$$

The solution is continuously extended at the $x = 0$ and $x = 1$ boundaries, and a terminal time of $t = 0.2$ is used. An exact solution is available via a Riemann solver. Although computations are done with a switch tolerance of $\epsilon_0 = 10^{-4}$; experiments reveal that $\epsilon_0 = 10^{-2}$ also works at the cost of introducing of small oscillations.

In Figure 3.7 numerical solutions of the ρ, u, P and Mach number M are plotted. A fixed mesh of $K = 200$ elements at $N = 1$ order is used. The discontinuities are

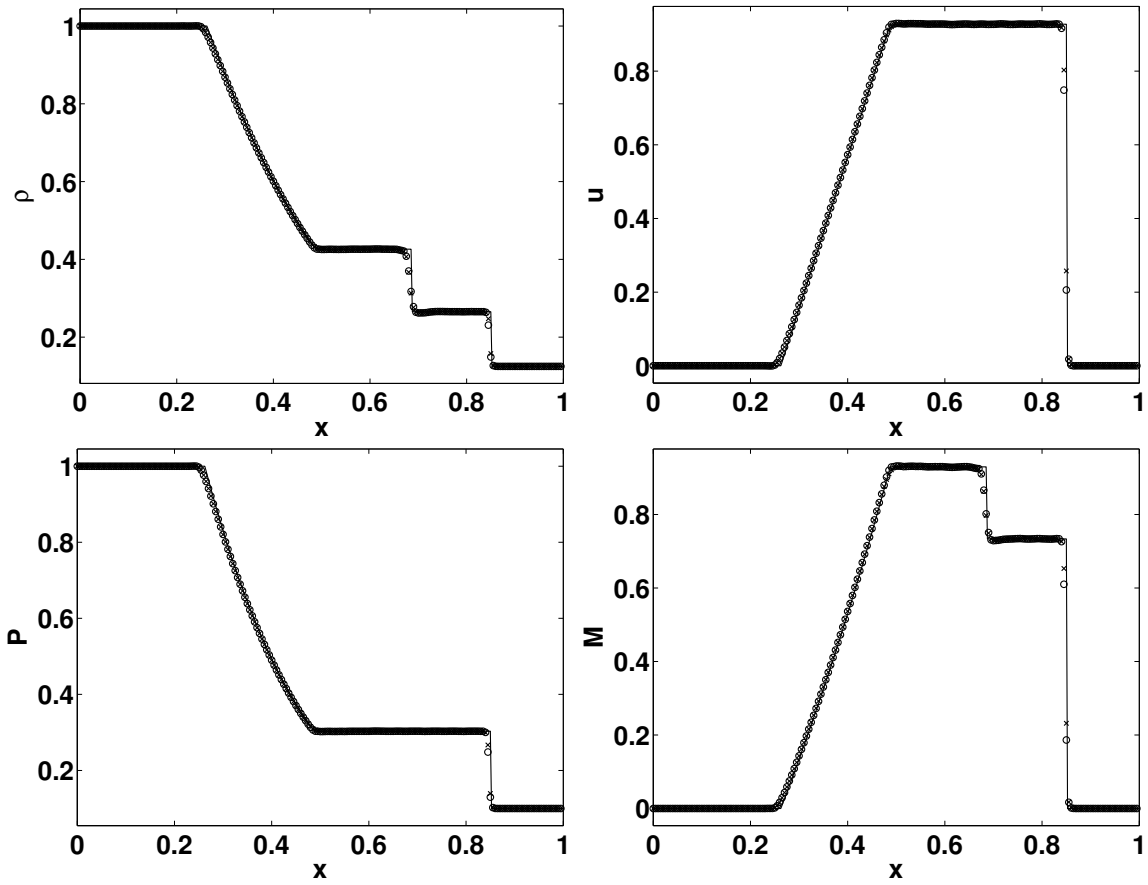

Figure 3.7: Numerical solution to Sod's problem (3.18). $K = 200$ elements and $N = 1$ order are used. Density ρ, velocity u, pressure P, and Mach number M are plotted. O's: Q-ENO reconstruction. X's: DK reconstruction. Solid line: exact solution.

all well-resolved without oscillations.

Results of the density ρ for varying order and number of elements are given in Figure 3.8. One observes that increasing both N and K further resolves the discontinuities and avoids oscillations.

Lax Problem

The Lax problem for the one-dimensional Euler equations (3.16) is given by the initial condition

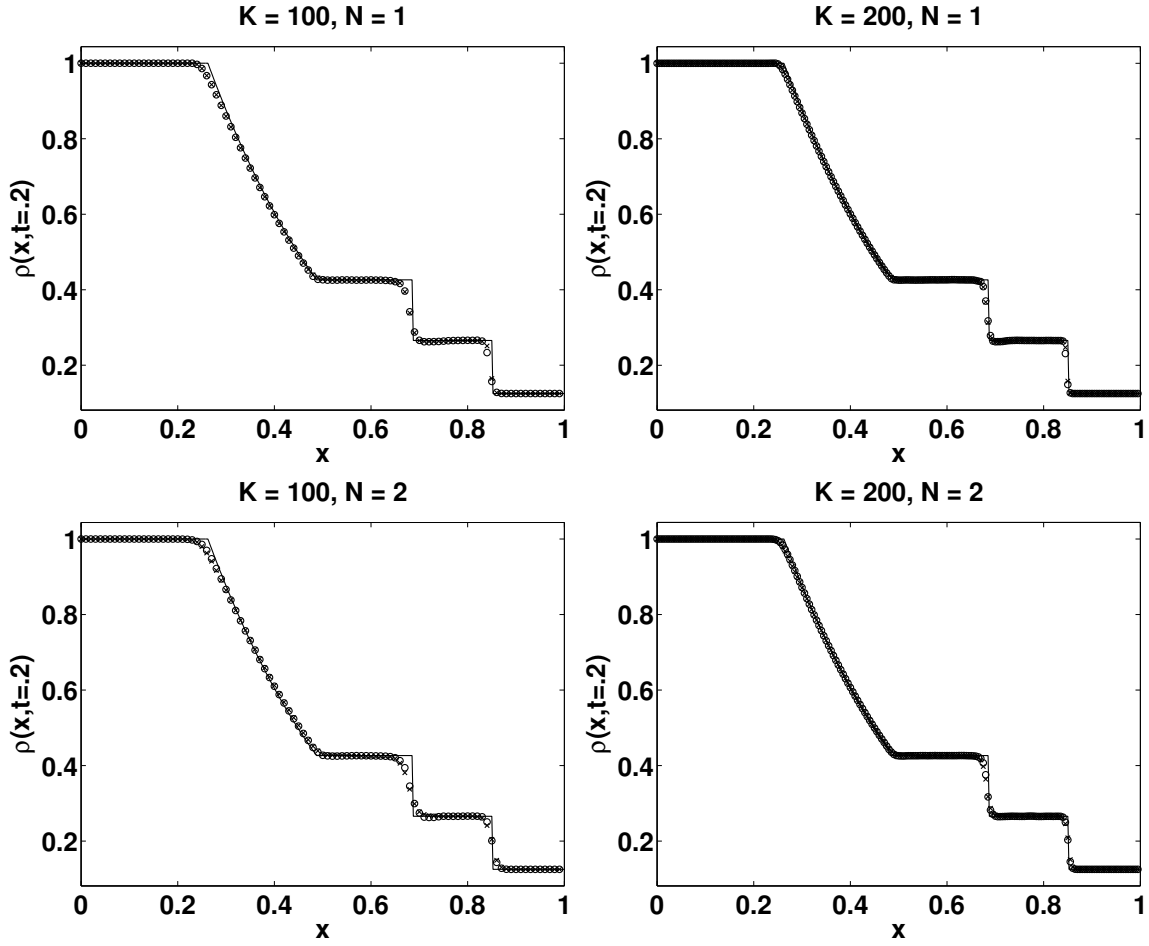

Figure 3.8: Numerical solution to Sod's problem (3.18). The total number of elements K and polynomial order N is varied. O's: Q-ENO reconstruction. X's: DK reconstruction. Solid line: exact solution.

$$(\rho, u, P)(x, 0) = \begin{cases} (0.445, 0.698, 3.528), & x \in [-5, 0) \\ (0.5, 0, 0.571), & x \in [0, 5] \end{cases} \tag{3.19}$$

The solution is continuously extended at the $x = -5$ and $x = 5$ boundaries. A terminal time of $t = 1.3$ is used. The Lax problem presents a greater challenge than the Sod problem (3.18) due to the shock region ($2 < x < 4$ at terminal time) initially having zero support. Solutions for a mesh of $K = 200$ elements at order $N = 2$ are given in Figure 3.9. An "exact" solution is provided by running the Hybrid with 4000 elements at order 2.

In Figure 3.10, a fixed $K = 200$ elements are used with varying $N = 1, 2, 3, 4$. The purpose of this simulation is to demonstrate the error introduced by using higher order solutions for the Lax problem. As N increases, the oscillations grow larger. With proper fine tuning of the parameters we may slightly decrease these oscillations, but generally speaking, for this problem it is best to increase K while keeping N low. To demonstrate this, in Figure 3.11 we consider a fixed $N = 1$ with varying $K = 100, 200, 400, 800$.

It is observed also that the DK scheme is more successful in preventing oscillations than Q-ENO.

Shock-Density Wave Problem

The Hybrid is applied to the difficult shock-density wave problem, with initial condition (3.20). This problem presents a considerable challenge: as the initial sine waves in the density grow in amplitude, we are at risk smearing the features with excessive dissipation as well as overshooting the correct solution due to aliasing errors.

$$(\rho, u, P)(x, 0) = \begin{cases} (3.857143, 2.629369, 10.333333) & x \in [-5, -4) \\ (1 + 0.2\sin(5x), 0, 1) & x \in [-4, 5] \end{cases} \tag{3.20}$$

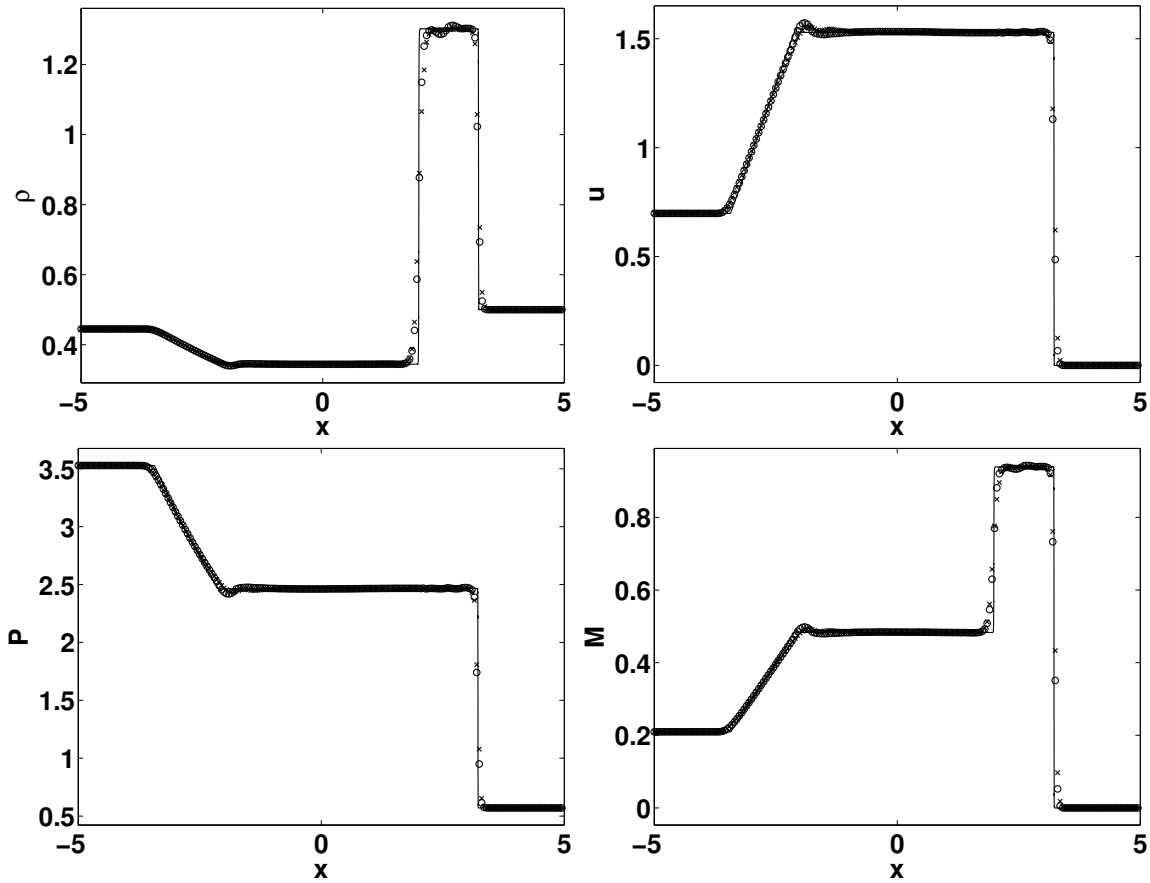

Figure 3.9: Numerical solution to the Lax problem (3.19). $K = 200$ elements and $N = 2$ order are used. Density ρ, velocity u, pressure P, and Mach number M are plotted. O's: Q-ENO reconstruction. X's: DK reconstruction. Solid line: exact solution.

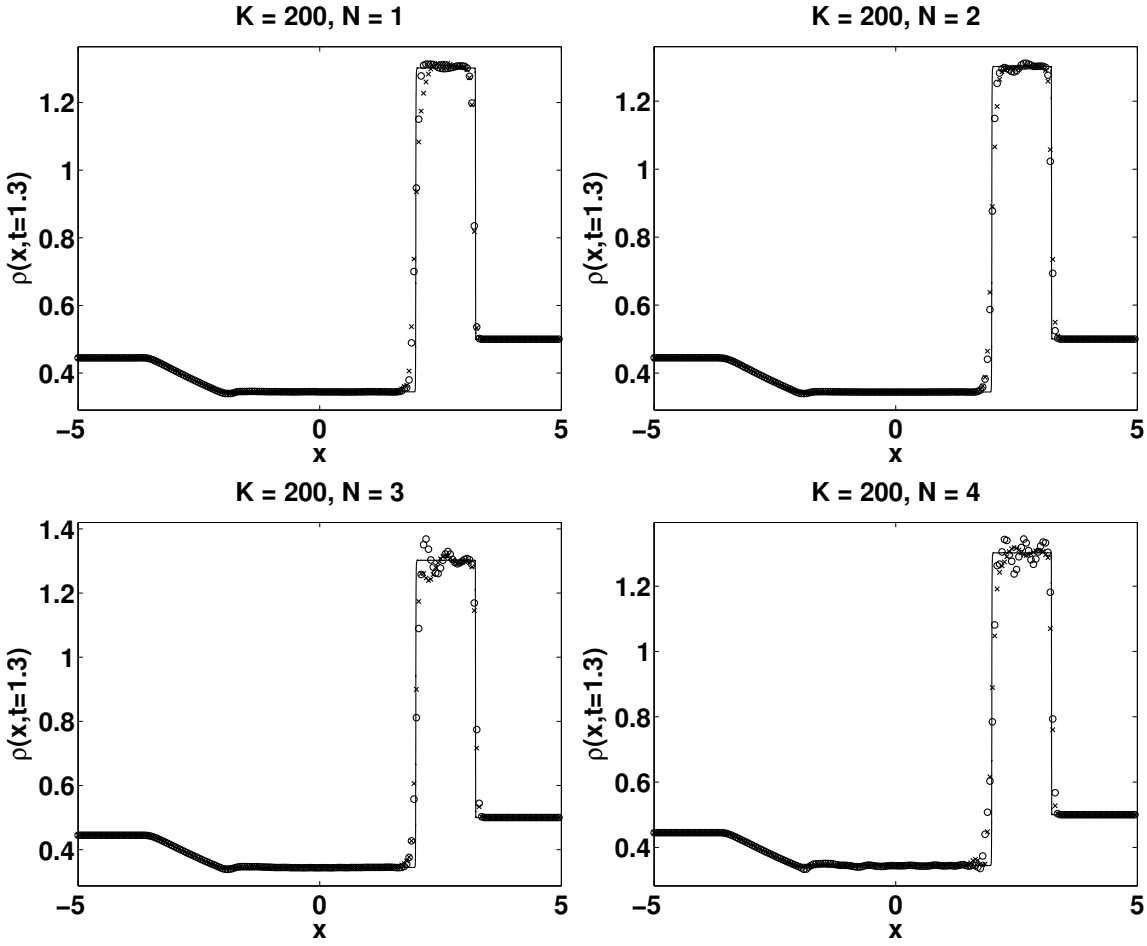

Figure 3.10: Numerical solution to the Lax problem (3.19). A fixed value of $K = 200$ grid points with varying $N = 1, 2, 3, 4$ order is given. O's: Q-ENO reconstruction. X's: DK reconstruction. Solid line: exact solution.

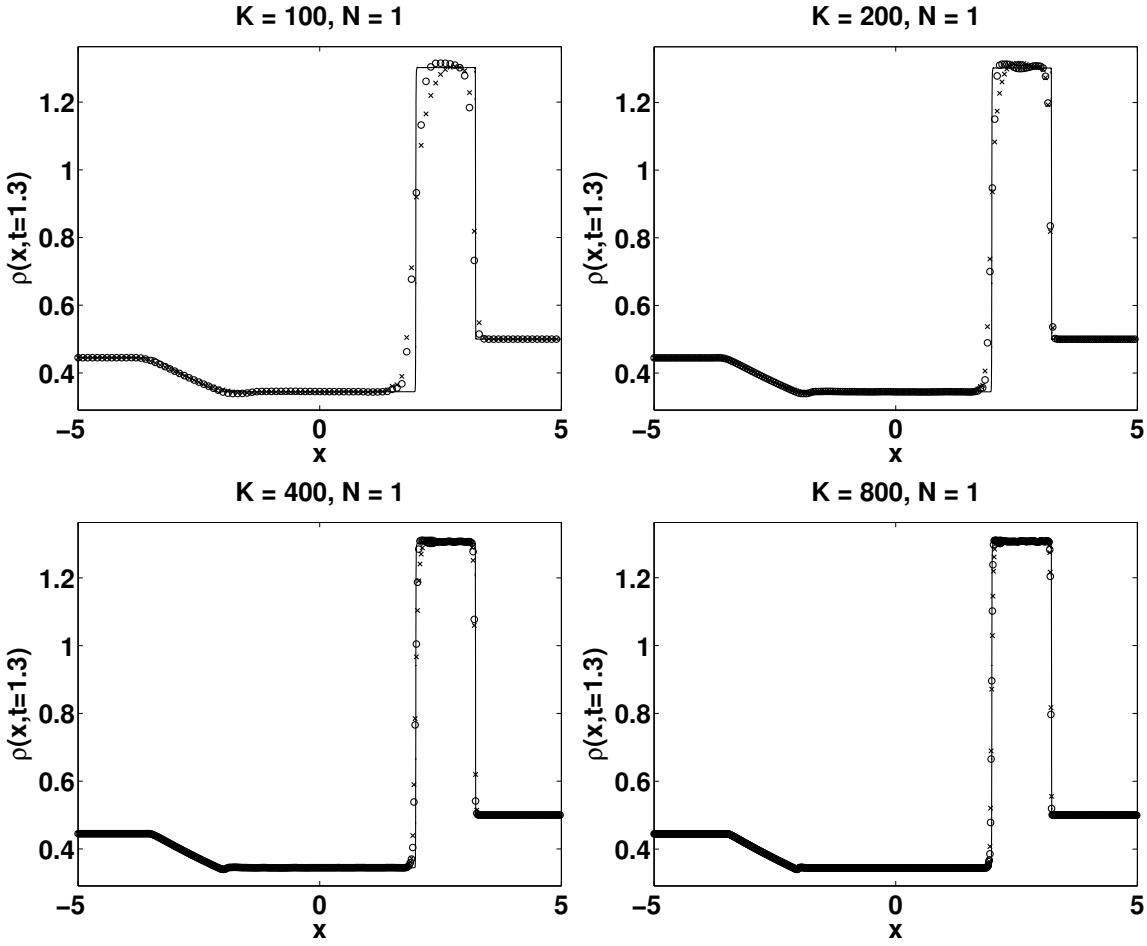

Figure 3.11: Numerical solution to the Lax problem (3.19). Solutions of order $N = 1$ and varying $K = 100, 200, 400, 800$ are given. O's: Q-ENO reconstruction. X's: DK reconstruction. Solid line: exact solution.

The solution is continuously extended at the $x = -5, x = 5$ boundaries. A terminal time of $t = 1.8$ is used. Numerical results for a mesh of $K = 400$ elements are given in Figure 3.12 at order $N = 3$ are given in Figure 3.12. An "exact" solution is provided by running the Hybrid with 4000 elements at order 2.

Numerical experiments reveal the choice of $K = 400$ roughly as a minimum for the number of elements needed to best resolve the solution. However, by increasing the switch tolerance to $\epsilon_0 = 10^{-2}$, and permitting the discontinuous Galerkin scheme to more often remain in place, resolution is achieved for lower K and higher N.

Examining the density ρ in Figure 3.12, we observe that the Q-ENO reconstruction appears to have slight over- and under-shoots in the naturally oscillatory regions corresponding roughly to $1 < x < 2.5$. This behavior grows worse for higher order, as is seen in Figure 3.13.

3.3 Two-Dimensional Observations

Thus far the Hybrid scheme has been presented only for one-dimensional problems. Unfortunately, when each scheme was extended to two dimensions, we were unable to robustly retain the ENO property. The main obstacle encountered appeared to be spurious oscillations introduced by demanding high-order polynomial reconstructions in regions where the underlying data was not sufficiently smooth.

Nevertheless, we may still verify the ability of each reconstruction to achieve high-order representation of smooth functions. We consider the smooth function

$$u(x, y) = \exp\left[-5\left(x^2 + y^2\right)\right], \quad (x, y) \in [-1, 1] \times [-1, 1]\,. \tag{3.21}$$

In Figure 3.14 three grids of increasing refinement ($K = 146, 568$, and 2310 triangular elements) are presented. For each mesh we reconstruct Eq. (3.21) from cell averages. In Table 3.5 convergence rates are reported by considering the decay of the L_1 error. These rates confirm the desired $O(h^{N+1})$ accuracy for reconstruction

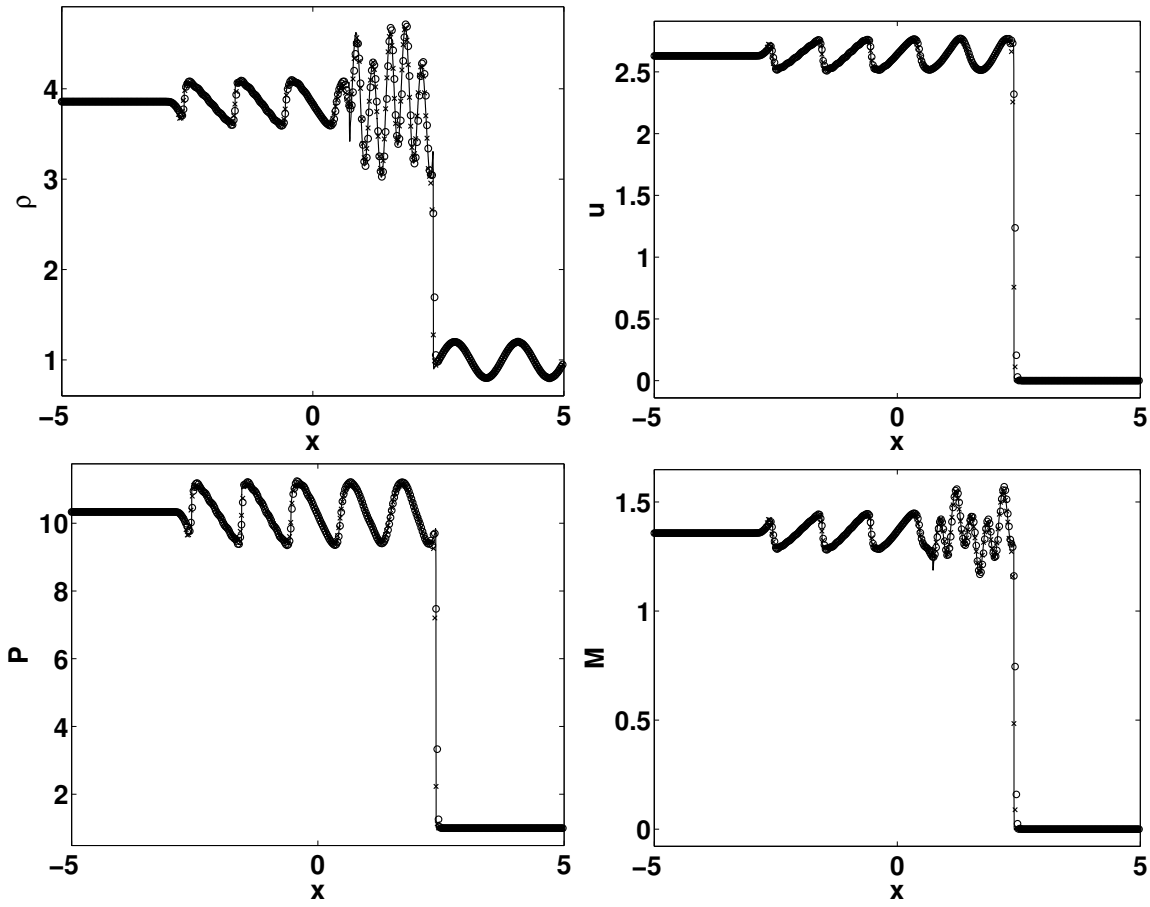

Figure 3.12: Numerical solution to the Shock density wave problem (3.20). $K = 400$ elements and $N = 3$ order are used. Density ρ, velocity u, pressure P, and Mach number M are plotted (one point per element). O's: Q-ENO reconstruction. X's: DK reconstruction. Solid line: exact solution.

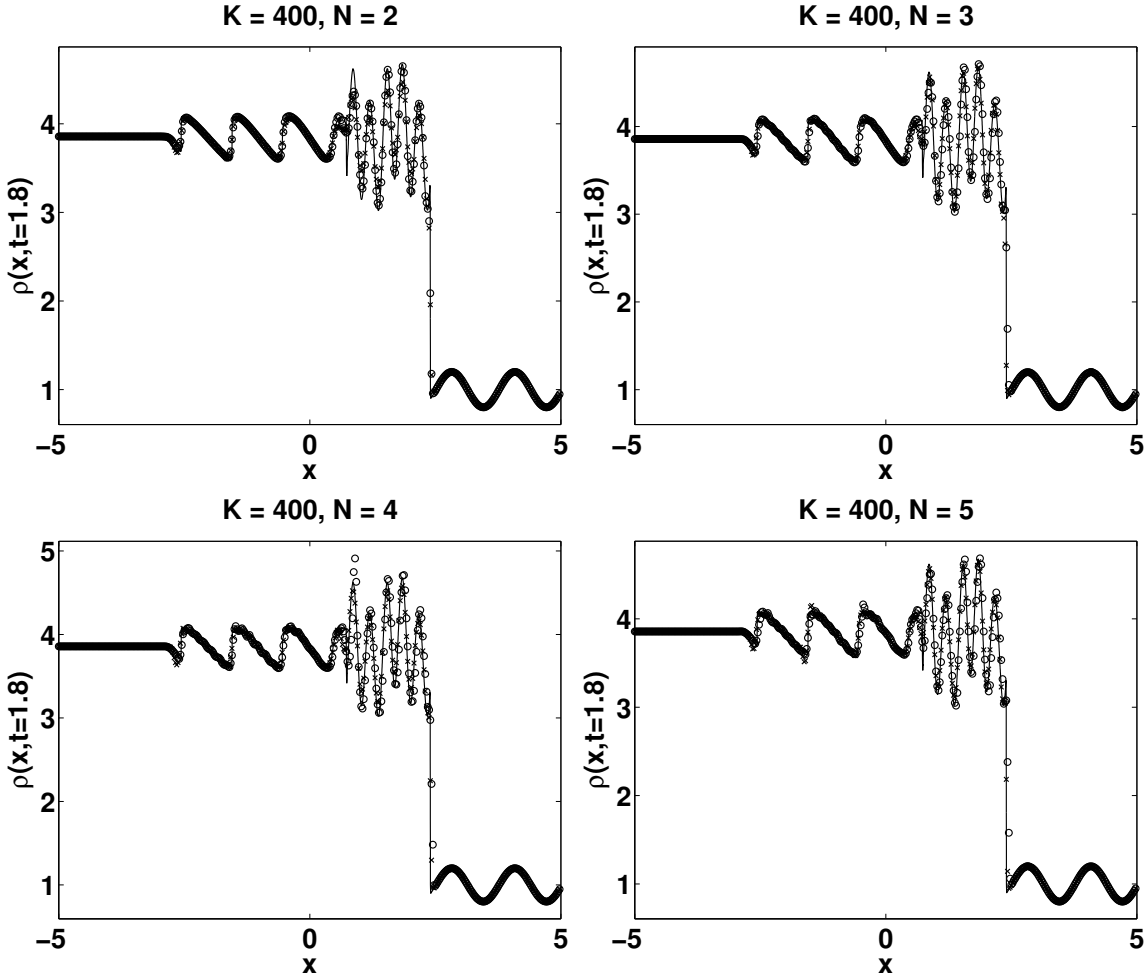

Figure 3.13: Numerical solution to the Shock density wave problem (3.20). $K = 400$ elements with varying order $N = 2, 3, 4, 5$ order are used. One point per element is plotted. O's: Q-ENO reconstruction. X's: DK reconstruction. Solid line: exact solution.

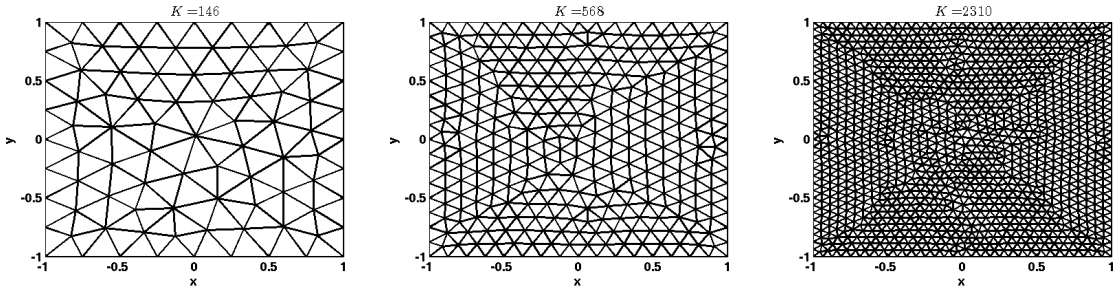

Figure 3.14: Sequence of refined grids with $K = 146, 568, 2310$ (unstructured) triangular elements.

		Q-ENO		DK	
	K	L_1-error	Convergence rate	L_1-error	Convergence rate
$N = 1$	146	1.39e-01		1.58e-01	
	568	3.06e-02	2.23	4.00e-02	2.01
	2310	8.64e-03	1.80	6.64e-03	2.56
$N = 2$	146	2.03e-02		2.99e-02	
	568	2.29e-03	3.21	3.37e-03	3.22
	2310	2.61e-04	3.10	2.51e-04	3.70
$N = 3$	146	1.30e-02		6.79e-03	
	568	6.14e-04	4.50	4.47e-04	4.00
	2310	3.74e-05	3.99	2.54e-05	4.09
$N = 4$	146	1.04e-02		4.86e-03	
	568	2.68e-04	5.39	1.50e-04	5.12
	2310	6.72e-06	5.25	4.13e-06	5.12

Table 3.5: Two-dimensional convergence study for DK and Q-ENO reconstructions for the function $u(x, y) = \exp[-5(x^2 + y^2)]$. L_1 error is given for K elements of order N. The schemes attain their desired reconstruction order.

of smooth functions, where we now take $h = 1/\sqrt{K}$.

3.4 Concluding Remarks

A Hybrid scheme consisting of the discontinuous Galerkin method conjoined to a finite volume method with high-order, ENO-class reconstruction algorithms for the computation of numerical flux, has been presented. A generalized slope limiter is used to detect regions of the solution where shocks may be present. Two reconstruc-

tion algorithms, QENO and DK, have been tested in the Hybrid. Unlike traditional ENO and WENO reconstructions, QENO and DK contain no requirement of a structured mesh. Numerical results have been presented for one-dimensional conservation laws which demonstrate the Hybrid's ability to accurately capture the solution while limiting non-physical oscillations. However, the reconstruction algorithms each failed to maintain monotonicity in the generalization to two-dimensional test problems on unstructured grids.

With successful results on nonlinear systems of conservation laws, albeit in the one-dimensional setting, we conclude that the general idea of the proposed Hybrid scheme has much merit, but work remains to be done to extend these successes to higher-dimensional problems. The key feature missing is a suitable reconstruction algorithm.

Part II

Reactive Infiltration Instability

Chapter 4

High-Order Methods for the Reactive Infiltration Instability in an Upwelling Compacting Mantle

The generation and segregation of melt from the earth's interior is a fundamental, but only partially understood, process. Melt is generated by adiabatic decompressional melting during upwelling in the Earth's upper mantle. Segregation of melt involves two-phase flow in which low viscosity melt percolates through a much more viscous solid matrix [53, 39, 48, 5]. During its upward migration via flow in porous media, melt generated in the deeper part of the upwelling mantle will interact both thermally and chemically with the overlying mantle. The main objective of the present study is to develop high-order numerical methods that can be used to study the chemical interaction between the melt and solid mantle during melt migration in an upwelling mantle column.

A fundamental geologic observation that gives insight into the existing theory of melt transport is that the chemical composition of erupted basalt is not in equilibrium with residual mantle at low pressure [40, 57, 19, 35, 34], particularly at diverging plate boundaries. Additionally, it is observed that the mantle is chemically and

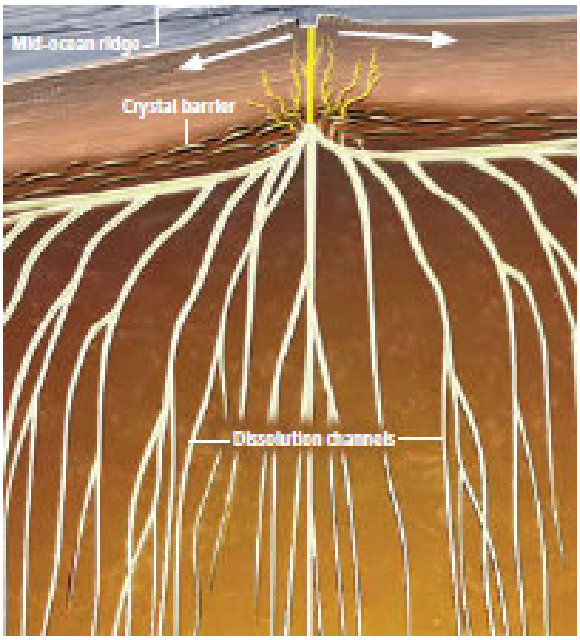

Figure 4.1: Illustration of the formation and coalescence of melt channels. Adiabatic decompressional melting occurs within the mantle. Channels migrate upwards, with impenetrable solid barriers guiding the melt towards the mid-ocean ridge. Picture taken from [31].

lithologically heterogeneous [32]. To preserve the geochemical signature developed at depth, melt must rise from depths of at least 30 km to the surface without extensive re-equilibration with the surrounding mantle. One such mechanism for this involves the reactive infiltration instability (RII) [8, 42, 2], where highly porous regions form due to a feedback between melt flow and dissolution. It has been suggested [42, 15, 33, 2, 1, 56] that high-permeability dunite channels act as conduits for focused flow, where melt may efficiently segregate from its source region while still maintaining its geochemical signature at depth. Numerical studies involving a simple finite difference method have been presented in [56, 55]; however, these results do explicitly not account for the soluble mineral orthopyroxene (opx), whose presence is integral to the formation of dunite channels. Moreover, they do not include an upwelling term and perform tests only for smaller Damköhler numbers.

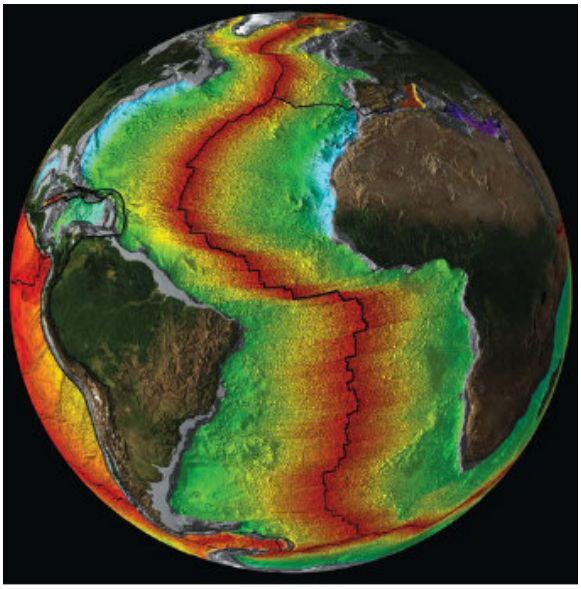

Figure 4.2: The Mid-Atlantic Ridge, a 10,000 km long string of volcanoes. Colors indicate relative age of the crust under the ocean. Red is youngest, with older ages as one approaches the continents. Picture taken from [31].

In this work we utilize the governing equations for reactive dissolution in a deformable, permeable medium presented in [2, 56], but with the addition of a solid upwelling term and an equation to track the abundance of opx in the matrix. A two-dimensional domain is considered, utilizing the discontinuous Galerkin finite element method in the vertical direction and a Fourier spectral method in the horizontal direction. An explicit fourth-order Runge-Kutta time discretization is used. Lateral variations in melt productivity, due to mantle heterogeneity, are represented by perturbations in the melt fraction at the base of the domain. It is observed that the formation and sustaining of melt channels in the simulation box are directly correlated to the perturbation prescribed.

In order to circumvent a restrictive time step posed by the reaction term, a formulation of the governing equations is presented where the concentration of the opx mineral is at equilibrium. Numerical solutions tentatively show agreement in

the equilibrium and kinetic formulations. A more sophisticated means of tracking of the regions with depleted opx fraction is necessary for accurate treatment of the mass transfer term.

4.1 Governing Equations

The governing equations are chosen from [56] and [39], with the addition of an equation to model the presence of the soluble mineral opx. The volume fraction of opx is important as geologists identify melt channel to be areas where opx has completely disappeared. We consider a three phase porous medium consisting of interconnected melt (f) and a matrix comprising of two minerals: olivine (ol) and orthopyroxene (opx). The mass conservation for phase $r \in \mathcal{S} = \{f, ol, opx\}$ is given as

$$\frac{\partial \left(\rho_r \phi_r\right)}{\partial t} + \nabla \cdot \left(\rho_r \phi_r \mathbf{V}_r\right) = \Gamma_r \tag{4.1}$$

where ϕ_r is the volume fraction, $\mathbf{V}_r$ is the phase velocity, and Γ_r is the bulk solid-to-fluid mass transfer rate of phase r, that is, the reaction rate. All phase fractions sum to unity:

$$\phi_f + \phi_{opx} + \phi_{ol} = 1 \tag{4.2}$$

We define c_r to be the concentration of soluble component in phase r, yielding the solute conservation equation

$$\frac{\partial}{\partial t} \sum_{r \in \mathcal{S}} \left(\rho_r c_r \phi_r\right) + \nabla \cdot \sum_{r \in \mathcal{S}} \left(\rho_r c_r \phi_r \mathbf{V}_r\right) = \nabla \cdot \sum_{r \in \mathcal{S}} \left(\rho_r \phi_r \mathbf{D}_r \nabla c_r\right), \tag{4.3}$$

where $\mathbf{D}_r$ is the diffusion tensor in phase r. The two solid phases *ol* and *opx* move with velocity

$$\mathbf{V}_{ol} = \mathbf{V}_{opx} = \mathbf{V}_s, \tag{4.4}$$

and the fluid velocity is denoted $\mathbf{V}_f$.

Conservation of momentum in the fluid is then given by the Darcy's law which relates $\mathcal{P}$, the pressure in excess of hydrodynamic pressure, to the velocities $\mathbf{V}_f, \mathbf{V}_s$. It is given as

$$\phi_f \left(\mathbf{V}_f - \mathbf{V}_s\right) = -\frac{\kappa_\phi}{\mu}\nabla\mathcal{P} \tag{4.5}$$

where μ is the viscosity of the fluid and κ_ϕ is the permeability, expressed as the power law

$$\kappa_\phi = \frac{d^2}{b}\phi_f^n. \tag{4.6}$$

Here, b is a constant and d is the mineral grain size. Both will disappear in the non-dimensionalization. n is an integer that is typically chosen as 2 or 3 for melt migration in the Earth's mantle; for this work we take $n = 3$.

The system is closed by accounting for conservation of momentum in the solid, given by the equation [39, 54]

$$\frac{\partial\mathcal{P}}{\partial x_i} = \frac{\partial}{\partial x_j}\eta\left(\frac{\partial\mathbf{V}_{si}}{\partial x_j} + \frac{\partial\mathbf{V}_{sj}}{\partial x_i}\right) + \frac{\partial}{\partial x_j}\left(\xi - \frac{2}{3}\eta\right)\nabla\cdot\mathbf{V}_s - \left(1-\phi_f\right)\Delta\rho g\delta_{i3}. \tag{4.7}$$

Here, η and ξ are the solid phase shear and bulk viscosities, respectively, while $\Delta\rho$ is the density difference between the solid and the melt.

In order to simplify the governing equations we make the following assumptions

1. A stoichiometric relationship β exists between *ol* produced and *opx* consumed by dissolution, so we can reduce the reaction rates Γ_r to the single degree of freedom Γ_{opx}:

$$\Gamma_{ol} = -\beta\Gamma_{opx} \tag{4.8}$$

$$\Gamma_f = -\left(\Gamma_{opx} + \Gamma_{ol}\right) = -\left(1-\beta\right)\Gamma_{opx} \tag{4.9}$$

2. With a uniformly upwelling mantle, the shear in the matrix is negligible, so we

take $\frac{\partial \mathbf{V}_{si}}{\partial x_j} = 0$ if $i \neq j$.

3. Solid mineral concentrations are constant and uniform, i.e., $c_{opx} = c^{eq}_{opx} = c^0_{opx}$ and $c_{ol} = c^{eq}_{ol} = c^0_{ol}$.

4. Diffusion in the solid is negligible, and isotropic in the fluid, so $\mathbf{D}_{ol} = \mathbf{D}_{opx} = 0$ and $\mathbf{D}_f = D$.

5. Porosity is very small ($\phi_f \ll 1$).

6. The phase densities ρ_r are constant and everywhere equal except in the buoyancy term $\Delta\rho$. We therefore absorb these into the reactive term,

$$\frac{\Gamma_r}{\rho_r} \to \Gamma_r \tag{4.10}$$

Using these assumptions, the mass conservation equations (4.1) become

$$\frac{\partial \phi_f}{\partial t} + \nabla \cdot (\phi_f \mathbf{V}_f) = -(1-\beta)\,\Gamma_{opx} \tag{4.11}$$

$$\frac{\partial \phi_{opx}}{\partial t} + \nabla \cdot (\phi_{opx} \mathbf{V}_s) = \Gamma_{opx} \tag{4.12}$$

$$\frac{\partial \phi_{ol}}{\partial t} + \nabla \cdot (\phi_{ol} \mathbf{V}_s) = -\beta\Gamma_{opx} \tag{4.13}$$

whereas the solute conservation equation (4.3) reduces to

$$\frac{\partial \left(\phi_f c_f + \phi_{opx} c^0_{opx} + \phi_{ol} c^0_{ol}\right)}{\partial t} + \nabla \cdot \left(\phi_f \mathbf{V}_f c_f + \phi_{opx} \mathbf{V}_s c^0_{opx} + \phi_{ol} \mathbf{V}_s c^0_{ol}\right) = \nabla \cdot (\phi_f D \nabla c_f)\,. \tag{4.14}$$

We assume a porosity-dependent bulk viscosity

$$\xi + \frac{4}{3}\eta = \frac{\xi_0}{\phi_f^m}, \tag{4.15}$$

where ξ_0 is a reference viscosity and the parameter $m \in [0, 1]$ allows for varying role

of the porosity in the bulk viscosity. This assumption, coupled with the simplification of neglecting matrix shear, reduces the momentum equation (4.7) to

$$\nabla \mathcal{P} = \nabla \left(\frac{\xi_0}{\phi_f^m} \nabla \cdot \mathbf{V}_s \right) - (1 - \phi_f) \Delta \rho g \mathbf{n}_z. \tag{4.16}$$

Substituting (4.16) into Darcy's law (4.5) yields

$$\phi_f (\mathbf{V}_f - \mathbf{V}_s) = -\frac{\kappa_\phi}{\mu} \left[\nabla \left(\frac{\xi_0}{\phi_f^m} \nabla \cdot \mathbf{V}_s \right) - (1 - \phi_f) \Delta \rho g \mathbf{n}_z \right]. \tag{4.17}$$

Recalling (4.2), we sum (4.11)-(4.13) and have

$$\nabla \cdot [\phi_f (\mathbf{V}_f - \mathbf{V}_s)] = -\nabla \cdot \mathbf{V}_s \tag{4.18}$$

Taking the divergence of (4.17) and equating it to (4.18) then yields the Helmholtz equation

$$\nabla \cdot \left[\frac{\kappa_\phi}{\mu} (\nabla p - (1 - \phi_f) \Delta \rho g \mathbf{n}_z) \right] = \frac{\phi_f^m p}{\xi_0}, \tag{4.19}$$

where we have introduced the compaction rate,

$$p = \frac{\xi_0}{\phi_f^m} \nabla \cdot \mathbf{V}_s. \tag{4.20}$$

Darcy's law (4.17) may be written as

$$\phi_f (\mathbf{V}_f - \mathbf{V}_s) = -\frac{\kappa_\phi}{\mu} [\nabla p - (1 - \phi_f) \Delta \rho g \mathbf{n}_z]. \tag{4.21}$$

Substituting (4.12)-(4.13) in to (4.14), we are left with a simplified equation for the evolution of the soluble material in the melt,

$$\frac{\partial (\phi_f c_f)}{\partial t} + \nabla \cdot (c_f \phi_f \mathbf{V}_f - \phi_f D \nabla c_f) = -\left(c_{opx}^0 - \beta c_{ol}^0 \right) \Gamma_{opx}. \tag{4.22}$$

Collectively, the conservation equations (4.11),(4.12), and (4.22), with the Helmholtz equation (4.19), and Darcy's law (4.21), form the system of equations for ϕ_f, ϕ_{opx}, c_f, and p. We need not solve for ϕ_{ol} because we solve for ϕ_f and ϕ_{opx} (Equation (4.2)). Also, we are not solving for the solid velocity $\mathbf{V}_s$ as it can be approximated by a leading uniform upwelling rate.

4.1.1 Reaction Rate

At this point, it is necessary to formally define the reaction term Γ_{opx}. From there the equations may be non-dimensionalized, so as to be most suitable for numerical simulation. Assuming an unequilibrated fluid concentration $\left(c_f \neq c_f^{eq}(\mathbf{x})\right)$, we take a first-order kinetic approximation

$$\Gamma_{opx} = -\alpha A(\phi_r)\left(c_f^{eq}(\mathbf{x}) - c_f\right). \tag{4.23}$$

The surface area term $A(\phi_r)$ is a function of the phases ϕ_r that moderates how much of the material is available for reaction, with reaction rate α. Introducing the parameter $l \in [0,1]$, we assume

$$A = \phi_{opx}^l. \tag{4.24}$$

The exponent l effectively allows us to control the role, if in any, played by the surface area ϕ_{opx} term. As will be seen later in a linear stability analysis, this term is significant when we examine the convergence of the kinetic to the equilibrium formulations. An alternative formulation given in Sec. 6.3 includes ϕ_f in A as well, although the presence of this term does not appear to have a significant effect. The equilibrium concentration is assumed to function of z only over a given length scale L:

$$c_f^{eq}(\mathbf{x}) = f(z) = c_f^0 + \frac{\Delta c}{L} z. \tag{4.25}$$

Taking a (lower) boundary at $z = 0$, $c_f^0 = c_f(z = 0)$ is the solubility at the base and Δc is the (constant) change in solubility over length L.

The most interesting setting is that of very high mass-fluid transfer rate, i.e., $\alpha \to \infty$. In this limit, the fluid concentration remains at equilibrium ($c_f = f(z)$), with instantaneous melt-solid reaction. An alternative reaction rate for this case, Γ_{opx}^{eq}, is obtained by substituting $c_f = c_f^{eq} = f(z)$ in (4.22) and (4.11). Rearranging terms, we have

$$\Gamma_{opx}^{eq} = \frac{D\nabla \cdot \left(\phi_f \nabla c_f^{eq}\right) - \phi_f \mathbf{V}_f \cdot \nabla c_f^{eq}}{c_{opx}^0 - \beta c_{ol}^0 - (1-\beta)\, c_f^{eq}} \tag{4.26}$$

With the assumption of a linearly-varying solubility gradient (4.25), we may simplify the ∇c_f^{eq} terms in (4.26). The reaction term may formally be given, therefore, as

$$\Gamma_{opx} = -\alpha\phi_{opx}\left(f(z) - c_f\right), \ \alpha < \infty \tag{4.27}$$

$$\Gamma_{opx}^{eq} = \frac{D\frac{\partial \phi_f}{\partial z} - \phi_f w}{L\hat{c}/\Delta c - (1-\beta)\, z}, \ \alpha = \infty \tag{4.28}$$

where $\hat{c} = c_{opx}^0 - \beta c_{ol}^0 - (1-\beta)\, c_f^{eq}(0)$ and $w = \mathbf{n}_z \cdot \mathbf{V}_f$, the vertical component of the fluid velocity.

4.1.2 Boundary Conditions

A two-dimensional simulation domain $(x, z) \in [0, x_L] \times [0, z_H]$ is taken, with the assumption of periodicity of the solution in the x direction. Boundary conditions in z are given as

$$
\begin{aligned}
p(0) &= 0, & \phi_f(0) &= \phi_f^0 \\
\phi_{opx}(0) &= \phi_{opx}^0, & c_f(0) &= c_f^0 \\
(\mathbf{n}_z \cdot \mathbf{V}_f)(0) &= w_0, & (\mathbf{n}_z \cdot \mathbf{V}_s)(0) &= W_0 \\
\left.\frac{\partial p}{\partial z}\right|_{z_H} &= 0, & \left.\frac{\partial c_f}{\partial z}\right|_{z_H} &= 0
\end{aligned}
\tag{4.29}
$$

4.2 Dimensionless Equilibrium Equations on a Reactive Time Scale

For a form more preferable for numerical simulation, we non-dimensionalize the governing equations according to the following (non-dimensionalized variables are denoted with a $*$):

$$
\begin{aligned}
(x, z) &= (Lx^*, Lz^*) & t &= Tt^* \\
\phi_f &= \phi_f^0 {\phi_f}^* & \phi_{opx} &= \phi_{opx}^0 {\phi_{opx}}^* \\
c_f &= c_f^0 + \Delta c {c_f}^* & p &= Pp^* \\
\kappa_\phi &= \kappa_0 {\kappa_\phi}^* & \mathbf{V}_s &= W_0 \mathbf{n}_z + \Delta V \tilde{\mathbf{V}}_s^* \\
\mathbf{V}_f &= w_0 \mathbf{V}_f^*
\end{aligned}
\tag{4.30}
$$

It is noted that the solid velocity $\mathbf{V}_s$ has been broken into two components: a uniform upwelling $W_0 \mathbf{n}_z$ and an additional piece $\Delta V \tilde{\mathbf{V}}_s^*$, where ΔV is assumed to be small.

δ	β	η_0	ϕ_f^0	Pe	Da	γ
10^{-2}	0.5	0.133	0.02	∞	$10^2 \to \infty$	$0 \to 10^{-2}$

Table 4.1: Typical values for the dimensionless parameters given in (4.32).

The length, pressure and time scales are chosen as

$$\begin{aligned} L &= \sqrt{\frac{\kappa_0 \xi_0}{\mu \left(\phi_f^0\right)^m}} & P &= \frac{\xi_0 \Delta V}{\left(\phi_f^0\right)^m L} \\ T &= \frac{L}{W_0 \delta} & \kappa_0 &= \frac{\left(\phi_f^0\right)^n d^2}{b} \\ w_0 &= \frac{\kappa_0 g \Delta \rho}{\phi_f^0 \mu} & \Delta V &= \phi_f^0 w_0 \delta. \end{aligned} \tag{4.31}$$

L is the compaction length [39]. Time adheres to a reactive scale evidenced by the δ in the denominator of T. The system is governed by the dimensionless parameters

$$\begin{aligned} \phi_f^0, \qquad & \eta_0 = \tfrac{\phi_f^0}{\phi_{opx}^0}, \qquad \delta = (1-\beta)\,\tfrac{\Delta c}{\hat{c}}, \\ Pe = \tfrac{w_0 L}{D}, \quad & Da = \tfrac{\alpha \hat{c} L \left(\phi_{opx}^0\right)^l}{\phi_f^0 w_0}, \qquad \gamma = \tfrac{W_0}{w_0} \end{aligned} \tag{4.32}$$

The γ parameter is most crucial as it effectively controls the rate of upwelling. Typical values for these parameters are given in Table 4.1.

Nondimensionalizing the governing equations according to the scales (4.31) yields the dimensionless equations

$$\frac{\partial \phi_f}{\partial t} = -\frac{\gamma}{\delta}\frac{\partial \phi_f}{\partial z} - \phi_f^0 \mathbf{V}_f \cdot \nabla \phi_f + \left(1 - \phi_f^0 \phi_f\right) p \phi_f^m - \Gamma_{opx} \tag{4.33}$$

$$\frac{\partial \phi_{opx}}{\partial t} = -\frac{\gamma}{\delta}\frac{\partial \phi_{opx}}{\partial z} - \phi_f^0 \mathbf{V}_f \cdot \nabla \phi_{opx} - \phi_f^0 \phi_{opx} \phi_f^m p + \eta \frac{\Gamma_{opx}}{1-\beta} \tag{4.34}$$

$$\delta \frac{\partial \left(\phi_f c_f\right)}{\partial t} = -\nabla \cdot \left(\phi_f c_f \mathbf{V}_f\right) + \frac{1}{Pe} \nabla \cdot \left(\phi_f \nabla c_f\right) - \Gamma_{opx} \tag{4.35}$$

$$-\nabla \cdot \left[\delta \phi_f^n \nabla p\right] + \delta p \phi_f^m = -\frac{\partial}{\partial z}\left[\left(1 - \phi_f^0 \phi_f\right) \phi_f^n\right] \tag{4.36}$$

From Darcy's law the fluid velocity is

$$\phi_f \left(\mathbf{V}_f - \gamma \phi_f \mathbf{n}_z - \phi_f^0 \mathbf{V}_s \right) = -\phi_f^n \left[\delta \nabla p - \left(1 - \phi_f^0 \phi_f \right) \mathbf{n}_z \right]. \tag{4.37}$$

To simplify matters, we drop terms of order ϕ_f^0 and the diffusive term in the fluid concentration equation. The following equations result, from which we will perform numerical simulations.

$$\frac{\partial \phi_f}{\partial t} = -\frac{\gamma}{\delta} \frac{\partial \phi_f}{\partial z} + p \phi_f^m - \Gamma_{opx} \tag{4.38}$$

$$\frac{\partial \phi_{opx}}{\partial t} = -\frac{\gamma}{\delta} \frac{\partial \phi_{opx}}{\partial z} + \eta \frac{\Gamma_{opx}}{1-\beta} \tag{4.39}$$

$$\delta \frac{\partial \left(\phi_f c_f \right)}{\partial t} = -\nabla \cdot \left(\phi_f c_f \mathbf{V}_f \right) - \Gamma_{opx} \tag{4.40}$$

$$-\nabla \cdot \left[\delta \phi_f^n \nabla p \right] + \delta p \phi_f^m = -\frac{\partial \phi_f^n}{\partial z} \tag{4.41}$$

$$\phi_f \mathbf{V}_f - \gamma \phi_f \mathbf{n}_z = -\phi_f^n \left(\delta \nabla p - \mathbf{n}_z \right) \tag{4.42}$$

Denoting the vertical component of fluid velocity $w = \mathbf{V}_f \cdot \mathbf{n}_z$, we may write the reaction rate as

$$\Gamma_{opx} = \begin{cases} -Da \phi_{opx}^l \left(z - c_f \right), Da < \infty \\ \frac{1}{1-\delta z} \left(\frac{1}{Pe_s} \frac{\partial \phi_f}{\partial z} - \phi_f w \right), Da = \infty \end{cases}. \tag{4.43}$$

The porosity evolution equation (4.38) was derived by rewriting the Darcy velocity term $\phi_f \mathbf{V}_f$ in Eq. (4.11), using (4.21). An alternative evolution equation, (still consistent with those presented) is derived by leaving the Darcy velocity in place,

$$\delta \frac{\partial \phi_f}{\partial t} + \nabla \cdot \left(\phi_f \mathbf{V}_f \right) = -\delta \Gamma_{opx}. \tag{4.44}$$

We will favor (4.38) over (4.44) for computational purposes, but (4.44) will be helpful later in Sec. 4.3 for computing the one-dimensional steady state.

Boundary conditions for (4.38)-(4.43) will be discussed below in Sec. 4.3.1.

4.3 One-Dimensional Steady State

In absence of an inflow perturbation, the steady-state solution to (4.38)-(4.43) is a function of z only. This steady state will form the basis of a linear stability analysis, as well as a starting point for transient simulations. One frequently occurring term is the volumetric fluid flux (also referred to as the Darcy velocity [4]),

$$q = \phi_f w. \tag{4.45}$$

With this new variable the steady state of the conservation equations (4.44),(4.39), and (4.40) gives rise to the ordinary differential equations

$$\frac{dq}{dz} = -\delta\Gamma_{opx}.\ q(0) = q_0 \tag{4.46}$$

$$\frac{d\phi_{opx}}{dz} = \eta\frac{\delta}{\gamma}\frac{\Gamma_{opx}}{1-\beta},\ \phi_{opx}(0) = 1 \tag{4.47}$$

$$\frac{dc_f}{dz} = \frac{\delta c_f - 1}{q}\Gamma_{opx},\ c_f(0) = c_f^0. \tag{4.48}$$

The equations (4.38) and (4.41), which govern the dynamics of compaction, yield the steady state equations

$$\gamma\frac{d\phi_f}{dz} - \delta p = -\delta\Gamma_{opx},\ \phi_f(0) = 1 \tag{4.49}$$

$$-\frac{d}{dz}\left[\phi_f^n\delta\frac{dp}{dz}\right] + \delta p = -\frac{d\phi_f^n}{dz},\ p(0) = p_0. \tag{4.50}$$

The boundary condition on the flux term q is given by (4.42) as

$$q_0 = \gamma - \left(\delta\frac{dp}{dz} - 1\right) \approx 1 + \gamma \tag{4.51}$$

Approximate solutions to (4.46)-(4.48) under the assumptions $\delta, \gamma \ll 1$, are

$$\bar{q} \approx \frac{q_0}{1-\delta z} - \frac{\delta q_0^2}{Da\,(1-\delta z)^4}\left[1 + \frac{\delta\eta q_0\left(c_f^0 + z\right)}{\gamma(1-\beta)(1-\delta z) - \delta\eta q_0(c_f^0 + z)}\right] \tag{4.52}$$

$$\bar{c}_f \approx z - \frac{q_0}{Da}\frac{1-\delta c_f^0}{(1-\delta z)^2}\left[1 - \frac{\delta\eta q_0(c_f^0 - z)}{\gamma(1-\beta)(1-\delta z) + \delta\eta q_0(c_f^0 - z)}\right] \tag{4.53}$$

$$\bar{\phi}_{opx} \approx 1 - \frac{\delta\eta q_0 z}{\delta(1-\beta)(1-\delta z)} + \frac{1}{Da}\frac{\delta\eta q_0^2}{\gamma(1-\beta)(\delta z-1)^4(1-\delta c_f^0)}\cdot \left[1 - \frac{\delta\eta q_0\left(z + c_f^0(1-\delta z)\right)}{-\gamma(1-\beta)(1-\delta z)(1-\delta c_f^0) + \delta\eta q_0\left(z + c_f^0(1-\delta z)\right)}\right] \tag{4.54}$$

The zeroth-order terms in these equations are the solutions to the ODE for the equilibrium formulation, demonstrating convergence as $Da \to \infty$ of the kinetic to equilibrium formulations. Approximate solutions to the compaction equations (4.49)-(4.50) are

$$\bar{\phi}_f \approx \left(1 + q_0\frac{\delta z}{1-\delta z}\right)^{1/n} \tag{4.55}$$

$$\bar{p} \approx \frac{-nq_0\phi_f^n}{(1-\delta z)^2\left(\gamma\phi_f + n\phi_f^n\right)} \approx -\frac{q_0}{(1-\delta z)^2} \tag{4.56}$$

4.3.1 Dimensionless Boundary Conditions

We would expect the non-dimensional form of the boundary conditions (4.29) to be

$$\begin{aligned} \phi_f(0) &= 1 & c_f(0) &= 0 \\ \phi_{opx}(0) &= 1 & p(0) &= 0 \end{aligned} \tag{4.57}$$

with Neumann outflow for p and c_f. The boundary condition $c_f(0) = 0$ presumes that the concentration is at equilibrium at in flow, while $p(0) = 0$ effectively prevents flux through the base. These combine to produce a sharp boundary layer near the inflow boundary, evidenced by steep gradients in ϕ_f and p. This steepens for increasing Da, although it reaches a critical value in the limit $Da \to \infty$.

The presence of such a boundary layer is somewhat artificial. In any event, it is no more arbitrary to assume a scenario absent of the boundary layer than it is with

one. Furthermore, the presence of steep gradients in the solution forces us to take higher resolution in the z direction than would otherwise be necessary. Accordingly, the boundary conditions for p and c_f may be reformulated so as to essentially remove this boundary layer. These modified boundary conditions computed by evaluating the steady states (4.53) and (4.56) at $z = 0$:

$$c_f(0) = -\frac{1+\gamma}{Da - \delta\,(1+\gamma)} \tag{4.58}$$

$$p(0) = -\frac{1+\gamma}{1-\delta z} \tag{4.59}$$

In Figure 4.3 an illustration is given which demonstrates the difference in the base states $\bar{p}$ and $\bar{c}_f$ depending upon which boundary condition is used.

4.3.2 Domain Length

The main goal of this project is to study the effect of that dunite channels have on the formation and migration of channels of melt. Dunite channels are marked by the evidence of previously melted opx. Therefore, a relevant domain length to consider is one that extends up to a region where the opx is exhausted. From the steady state (4.54) this is closely approximated

$$z_H = \delta^{-1}\left(1 - \frac{q_0}{q_0 + \frac{\gamma(1-\beta)}{\eta}}\right) \tag{4.60}$$

where as before, $q_0 = 1 + \gamma$. Typically we take a domain height slightly shorter ($\sim$ 75%) than prescribed in (4.60); when heterogeneity is introduced into the simulation, the height of opx exhaustion will be less than z_H due to the formation of melt channels.

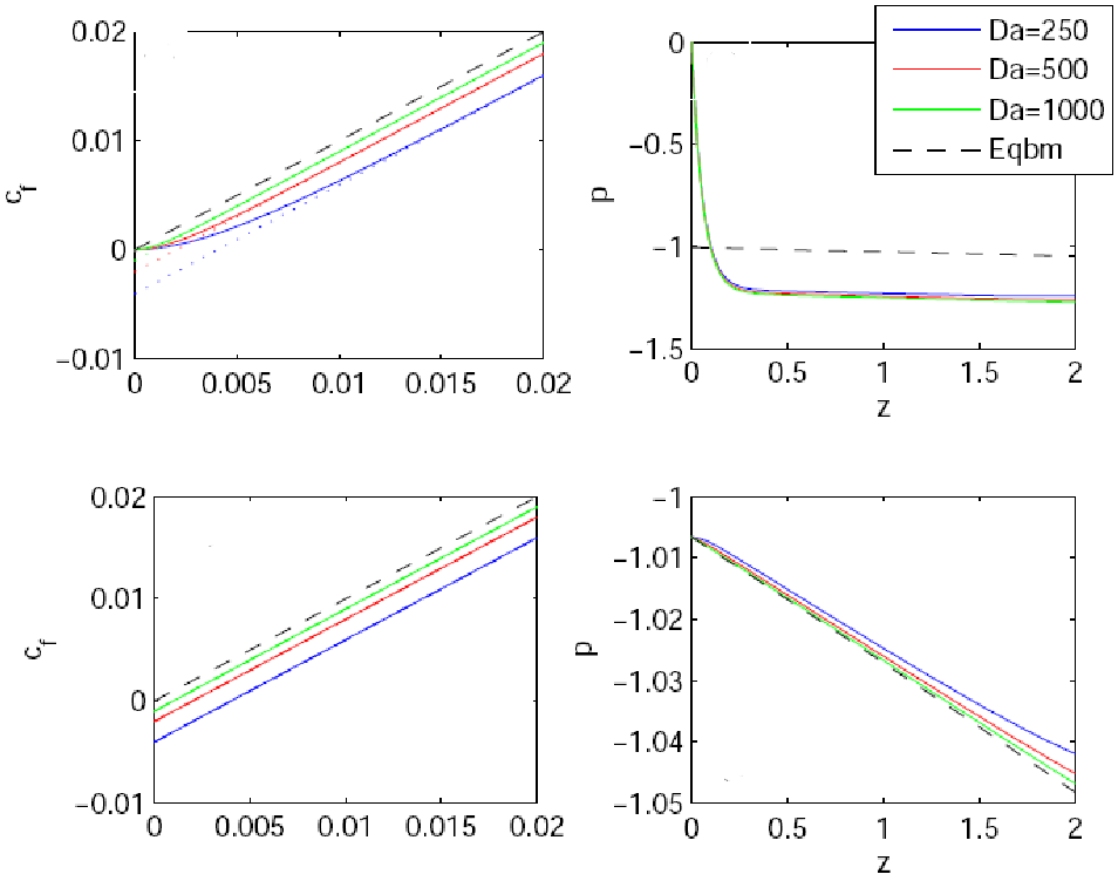

Figure 4.3: Base states $\bar{p}, \bar{c}_f$ for the difference choice of boundary conditions. Top: $p(0) = c_f(0) = 0$. Bottom: modified boundary conditions as given in Eq. (4.58). The domain for c_f is shortened to $z \in [0, 0.02]$, as outside of this it is essentially parallel to $c_f = c_f^{eq} = z$.

4.4 Numerical Implementation

We perform numerical simulation on the dimensionless equations (4.38)-(4.43). In the vertical direction, we implement a discontinuous Galerkin (DG) finite element method, while in the horizontal direction, we implement a Fourier spectral method. Each of these methods allow for high-order accuracy due to the large basis from which the solutions $\phi_f, \phi_{opx}, c_f, p$ are chosen.

In the discussion to follow, we outline the numerical scheme by presenting details of the solution representation, giving examples of the differentiation matrices in each coordinate direction. A fair amount of discussion is given for how the grid points in the DG direction are selected, necessary for the nodal implementation of the different DG operators. We will present the semidiscrete numerical scheme, followed by additional remarks including time integration and filtering for nonlinear stability.

4.4.1 Fourier Spectral Discretization

The scheme is discretized in the x-direction by a Fourier spectral method,

$$u_h(x,\cdot,t) = \sum_{i=0}^{M-1} u(x_i,\cdot,t) l_i^{FS}(x), \quad x \in [0, x_L]\,. \tag{4.61}$$

We assume an even number M of equally spaced horizontal grid points

$$x_i = \tfrac{i}{M} x_L, \quad i = 0, \ldots, M-1, \tag{4.62}$$

corresponding to the Lagrange interpolating polynomial

$$l_i^{FS}(x) = \frac{1}{M} \sin\left[\frac{\pi N}{x_L}(x - x_i)\right] \cot\left[\frac{\pi}{x_L}(x - x_j)\right]. \tag{4.63}$$

The (matrix) differentiation operator is given as

$$\left(\mathcal{D}^x\right)_{ij} = \frac{d}{dx} l_j^{FS}(x)\Big|_{x_i} = \begin{cases} (-1)^{i+j} \frac{\pi}{x_L} \cot\left[\frac{\pi}{x_L}(x_i - x_j)\right], & i \neq j \\ 0, & i = j. \end{cases} \tag{4.64}$$

Typically one would compute derivatives with a fast Fourier transform (FFT), but in our case this is insufficient due to the nonlinearity in the Helmholtz equation (4.19).

4.4.2 Discontinuous Galerkin Discretization

Much of the background detail of the discontinuous Galerkin method has already been presented in Sec. 1.3. Here we will detail more specifically implementation issues.

Building operators for the discontinuous Galerkin method is a bit more complex than the Fourier spectral method, as corresponding matrices are not written in such an easy, closed-form as in (4.64). To build the DG operators we first present a small amount of background material concerning orthogonal polynomials. From this we may compute the grid points, providing a basis for which we can build the nodal-based operators.

Jacobi Polynomials

To begin with, we recall the definition of the classical Jacobi polynomial $P_n^{(\alpha,\beta)}(\xi)$ of order n as the solution to the singular Sturm-Liouville eigenvalue problem

$$\frac{d}{d\xi}\left(1-\xi^2\right) w(\xi) \frac{d}{d\xi} P_n^{(\alpha,\beta)}(\xi) + n\left(n+\alpha+\beta+1\right) w(\xi) P_n^{(\alpha,\beta)}(\xi) = 0 \tag{4.65}$$

where we are in the (one-dimensional) reference coordinate system $\xi \in [-1, 1]$. The weight function w is given as

$$w(\xi) = (1-\xi)^\alpha (1-\xi)^\beta. \tag{4.66}$$

Multiplying (4.65) through by the polynomial $P_m^{(\alpha,\beta)}$ and integrating, we easily arrive at the orthogonality of the family of polynomials. We furthermore assume a normalization so that they are orthonormal,

$$\int_{-1}^{1} P_m^{(\alpha,\beta)}(\xi) P_n^{(\alpha,\beta)}(\xi) w(\xi) d\xi = \delta_{mn}. \tag{4.67}$$

A special case is $\alpha = \beta = 0$, resulting in the Legendre polynomials $P_n^{(0,0)}$ whose weight function is the identity. There is no known way of simply evaluating the Jacobi polynomials, but it may be done via the recurrence relationship

$$\xi P_n^{(\alpha,\beta)}(\xi) = a_n P_{n-1}^{(\alpha,\beta)}(\xi) + b_n P_n^{(\alpha,\beta)}(\xi) + a_{n+1} P_{n+1}^{(\alpha,\beta)}(\xi), \tag{4.68}$$

where the coefficients are given as

$$a_n = \frac{2}{2n+\alpha+\beta} \sqrt{\frac{n(n+\alpha+\beta)(n+\alpha)(n+\beta)}{(2n+\alpha+\beta-1)(2n+\alpha+\beta+1)}} \tag{4.69}$$

$$b_n = -\frac{\alpha^2 - \beta^2}{(2n+\alpha+\beta)(2n+\alpha+\beta+2)}. \tag{4.70}$$

The initial values of the recurrence are available, with $\Gamma(\cdot)$ as the standard Gamma function:

$$P_0^{(\alpha,\beta)}(\xi) = \sqrt{2^{-\alpha-\beta-1} \frac{\Gamma(\alpha+\beta+2)}{\Gamma(\alpha+1)\Gamma(\beta+1)}} \tag{4.71}$$

$$P_1^{(\alpha,\beta)}(\xi) = \frac{1}{2} P_0^{(\alpha,\beta)}(\xi) \sqrt{\frac{\alpha+\beta+2}{(\alpha+1)(\beta+1)}} ((\alpha+\beta)\xi + (\alpha-\beta)). \tag{4.72}$$

Lastly, we mention a useful property of the Jacobi polynomials [58]

$$\frac{d}{d\xi} P_n^{(\alpha,\beta)}(\xi) = \sqrt{n(n+\alpha+\beta+1)} P_{n-1}^{(\alpha+1,\beta+1)}(\xi). \tag{4.73}$$

The reason we concern ourselves with this discussion of Jacobi Polynomials

is found in a well known [16] result connecting Jacobi polynomials and Gaussian quadratures for the approximation of integrals. It can be shown that

$$\int_{-1}^{1} f(\xi)w(\xi)d\xi = \sum_{i=1}^{N+1} f(\xi_i)w_i \tag{4.74}$$

is exact for polynomials f of degree $2N+1$ if we choose ξ_i to be the roots of $P_{N+1}^{(\alpha,\beta)}(\xi)$ with corresponding weights w_i. To compute ξ and w_i we use the recurrence relationship (4.68). Setting $P_{N+1}^{(\alpha,\beta)}(\xi_i) = 0$ results in a symmetric tridiagonal eigenvalue problem whose eigenvalues are the Gauss points ξ_i; the weights w_i may be recovered from the eigenvectors (see [21] for more details).

Selecting the DG Nodes

All of the machinery presented is utilized in computing the Gauss-Lobatto points, for which we use to represent the solution. For simplicity we present the solution as represented on the reference element $\xi \in [-1, 1]$. A linear transformation then connects ξ to the physical domain $z \in [0, z_H]$. This approach may be expanded to multiple DG elements by simply keeping track of the Jacobians of each (reference to physical) transformation.

Although above we have presented a method to evaluate the Gauss points, these points are unsatisfactory as they do not include the end grid points $\xi = -1, 1$. We prefer to have a nodal representation at these locations in order to more easily implement boundary conditions. Instead, for an element of solution order N, we select the $N+1$ Gauss-Lobatto nodes. These correspond to the roots of the polynomial

$$\left(1-\xi^2\right)\frac{d}{d\xi}P_N^{(0,0)}\left(\xi\right).$$

Recalling Equation (4.73), we see that the Gauss-Lobatto points are the $N-2$ order Gauss points of $P_{N-2}^{(\alpha+1,\beta+1)}$ with the end points $\xi = -1, 1$ added.

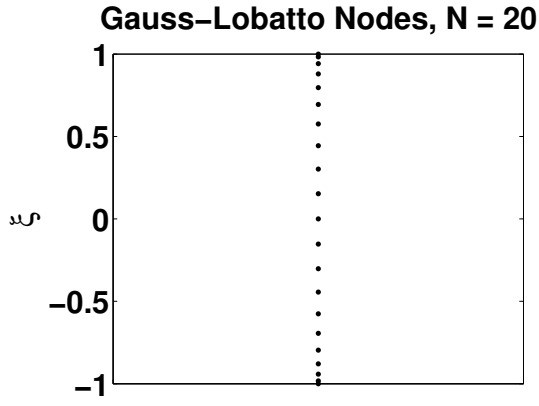

Figure 4.4: Gauss-Lobatto quadrature nodes for order $N = 20$. The nodes are given over the one-dimensional reference element $\xi \in [-1, 1]$.

The Gauss-Lobatto nodes cluster as $O(1/N^2)$ near the boundaries. An illustration of this clustering is given in Figure 4.4 for the value $N = 20$.

The DG Differentiation Matrix

Deriving the discontinuous Galerkin derivative operator requires a bit more effort than than was needed for the Fourier differentiation matrix (4.64). In the discussion that follows we compute the local DG differentiation matrix. Unless otherwise stated, all polynomials P_n are assumed to be Legendre polynomials, that is,

$$P_n(\xi) := P_n^{(0,0)}(\xi).$$

Consider the nodal representation of an order N polynomial

$$u(\xi) = \sum_{j=1}^{N+1} u(\xi) l_j^{DG}(\xi), \quad \xi \in [-1, 1] \tag{4.75}$$

with Lagrange interpolating polynomials l_j^{DG} based on the $N+1$ Gauss-Lobatto points. We seek a matrix $\mathcal{D}^\xi$,

$$\frac{\partial u}{\partial \xi} = \sum_{j=1}^{N+1} u(\xi) \frac{\partial l_j^{DG}}{\partial \xi} = \mathcal{D}^\xi \mathbf{u}, \tag{4.76}$$

where the vector $\mathbf{u} = (u(\xi_1), ..., u(\xi_{N+1}))^T$. The entries of the matrix can be written, then, as

$$\left(\mathcal{D}^\xi\right)_{ij} = \left. \frac{dl_j^{DG}}{d\xi} \right|_{\xi_i}. \tag{4.77}$$

To compute the entries of $\mathcal{D}^\xi$ we recall the Vandermonde matrix introduced in Eq. (1.7), with entries $(\mathcal{V})_{ij} = P_{j-1}(\xi_i)$. As a consequence of the uniqueness of polynomial interpolation, we have

$$\mathcal{V}^T \mathbf{l}(\xi) = \mathbf{P}(\xi), \tag{4.78}$$

where $\mathbf{l}(\xi) = (l_1(\xi), ..., l_{N+1}(\xi))^T$ and $\mathbf{P} = (P_0(\xi), ..., P_N(\xi))^T$. From Eq. (4.78) we differentiate with respect to ξ to get

$$\mathcal{V}^T \frac{d}{d\xi} \mathbf{l}(\xi) = \frac{d}{d\xi} \mathbf{P}(\xi). \tag{4.79}$$

Combining (4.77) and (4.79), we have

$$\mathcal{V}^T \left(\mathcal{D}^\xi\right)^T = \left(\mathcal{V}^\xi\right)^T, \tag{4.80}$$

where the matrix $\mathcal{V}^\xi$ is defined as

$$\left(\mathcal{V}^\xi\right)_{ij} = \left. \frac{dP_j}{d\xi} \right|_{\xi_i}. \tag{4.81}$$

To compute the entries in (4.81) we use the identity

$$\frac{dP_n^{(0,0)}}{d\xi} = \sqrt{n(n+1)}P_{n-1}^{(1,1)}(\xi) \tag{4.82}$$

along with the recurrence relationships from Eq. (4.68). With $\mathcal{V}^\xi$ and $\mathcal{V}$, we may now compute $\mathcal{D}^\xi$ from Eq. (4.80),

$$\mathcal{D}^\xi = \mathcal{V}^\xi \mathcal{V}^{-1} \tag{4.83}$$

The $\mathcal{D}^\xi$ matrix presented operates on the reference coordinate ($\xi \in [-1, 1]$), and so the corresponding matrix in physical space ($z \in [0, z_H]$) includes an additional factor from the Jacobian. For a DG element of length h this matrix is

$$\mathcal{D}^z = \frac{\partial z}{\partial \xi}\mathcal{D}^\xi = \frac{2}{h}\mathcal{D}^\xi. \tag{4.84}$$

The matrix $\mathcal{D}^z$ presented above is only part of the story when it comes to the DG differentiation operator. A complete discussion involves the numerical flux.

Mass and Stiffness Matrices

Later in Sec. 4.4.4 we will present the full implementation of the scheme. There we will use $\mathcal{M}^\xi$ and $\mathcal{S}^\xi$, the (local) mass and stiffness matrices, respectively. These are defined as

$$\left(\mathcal{M}^\xi\right)_{ij} = \int_{-1}^{1} l_i^{DG}(\xi) l_j^{DG}(\xi) d\xi \tag{4.85}$$

where l are the standard Lagrange interpolating polynomials for the selected set of nodes on which the DG solution is represented, and

$$\left(\mathcal{S}^\xi\right)_{ij} = \int_{-1}^{1} l_i^{DG}(\xi) \frac{dl_j^{DG}(\xi)}{d\xi} d\xi. \tag{4.86}$$

It can be shown that these matrices are computed easily by the identities

$$\mathcal{M}^{\xi} = \left(\mathcal{V}\mathcal{V}^{T}\right)^{-1} \tag{4.87}$$

and

$$\mathcal{S}^{\xi} = \mathcal{M}^{\xi}\mathcal{D}^{\xi}. \tag{4.88}$$

As will be seen in the implementation, the stiffness matrix $\mathcal{S}^{\xi}$ typically is replaced with the differentiation matrix $\mathcal{D}^{\xi}$, or its physical space representation, $\mathcal{D}^{z}$. A similar relationship between reference and physical space exists with the mass matrix. For a DG element of length h, we define the mass matrix in physical space, $\mathcal{M}^{z}$, as

$$\mathcal{M}^{z} = \frac{h}{2}\mathcal{M}^{\xi} = \frac{h}{2}\left(\mathcal{V}\mathcal{V}^{T}\right)^{-1} \tag{4.89}$$

We go to the effort of computing the matrix $\mathcal{D}^{\xi}$ in reference space, and then transforming to physical space at the end, because in a discontinuous Galerkin discretization the solution is defined piecewise on elements. Therefore we build all DG operators in the setting of the reference element, invoking the transformations as necessary.

4.4.3 Representing the Solution

Combining the horizontal and vertical representations of the solution (Eqs. (4.61) and (4.75), respectively), solutions on the k^{th} DG element are represented as

$$u_h^k(x,z,t) = \sum_{i=0}^{M-1}\sum_{j=0}^{N} u_h^k(x,z,t) l_i^{FS}(x) l_j^{k,DG}(z) \tag{4.90}$$

where $l_i^{FS}, l_j^{k,DG}$ are the Lagrange interpolating polynomials associated with Fourier nodes x_i and Gauss-Lobatto nodes z_j^k, respectively.

The computational domain is the (rectangular) tensor product of the Fourier

nodes $\{x_i\}$ and Gauss-Lobatto nodes $\left\{z_j^k\right\}$. For implementation purposes, it has been useful to treat the solutions as two-dimensional arrays. In the representation (4.90) this is visualized as

$$u_h^k(x,z,t) = \boldsymbol{u} = \left\{ \begin{Bmatrix} u_{0,0} \\ u_{1,0} \\ \vdots \\ u_{N,0} \end{Bmatrix}, \begin{Bmatrix} u_{0,1} \\ u_{1,1} \\ \vdots \\ u_{N,1} \end{Bmatrix}, \cdots \begin{Bmatrix} u_{0,M-1} \\ u_{1,M-1} \\ \vdots \\ u_{N,M-1} \end{Bmatrix} \right\} \tag{4.91}$$

where $u_{j,i} = u_h^k(x_i, z_j^k, t)$. In such an implementation, operators in z act on columns from the left while operators in x act on rows from the right. For example, derivatives in the x and z directions are discretized as

$$\frac{\partial u}{\partial x} \simeq \mathbf{u}\left(\mathcal{D}^x\right)^T, \quad \frac{\partial u}{\partial z} \simeq \mathcal{D}^z \mathbf{u} \tag{4.92}$$

A global solution (that is, one containing all DG elements) is arranged by vertically concatenating the solution arrays. In this way, global DG operators (e.g. the mass matrix, used in the Helmholtz equation discretization (4.104)) are constructed by considering a sparse, block-diagonal arrangement of each element's local operator.

4.4.4 Spatial Discretization

Prior discussion in Chapter 1 generally outlined the discontinuous Galerkin method for a hyperbolic conservation law, but avoided implementation-specific details. Here we will step through the implementation for the system (4.38)-(4.43).

The previously introduced notations for the physical-space mass and differentiation matrices is broadened to account for varying DG element sizes. For a given element k of length h^k, the notation for these is

$$\mathcal{M}^{(z,k)}, \quad \mathcal{D}^{(z,k)}$$

so as to denote that they are z-operators applied to the k^{th} discontinuous Galerkin element.

We proceed using the strong form of the discontinuous Galerkin method, introduced previously in Eq. (1.11). We will evolve the solution on DG element k, corresponding to physical space $z \in \left[z_B^k, z_T^k\right]$. In this case the discrete solution quantities are represented as

$$(\phi_f, \phi_{opx}, c_f, p) \simeq (\boldsymbol{\phi_f}, \boldsymbol{\phi_{opx}}, \boldsymbol{c_f}, \boldsymbol{p})\,. \tag{4.93}$$

In the outline to follow, we attempt to motivate a matrix-vector implementation, invoking the matrices $\mathcal{M}^{(z,k)}, \mathcal{D}^{(z,k)}$, and $\mathcal{D}^x$ to act on solution vectors (bold-faced quantities; e.g. those in Eq. (4.93)). When consecutive vectors appear, products are computed entry-wise (i.e. on the nodes).

Consider the porosity evolution equation (4.38). The local semidiscrete scheme is

$$\begin{aligned}\mathcal{M}^{(z,k)}\frac{d\boldsymbol{\phi_f}}{dt} =& -\frac{\gamma}{\delta}\mathcal{S}^{\xi}\boldsymbol{\phi_f} + \left[\boldsymbol{l}^k(z)\left(\frac{\gamma}{\delta}\boldsymbol{\phi_f} - \left(\frac{\gamma}{\delta}\boldsymbol{\phi_f}\right)^*\right)\right]_{z_B^k}^{z_T^k} \\ &+ \mathcal{M}^{(z,k)}\boldsymbol{p}{\boldsymbol{\phi_f}}^m - \mathcal{M}^{(z,k)}\boldsymbol{\Gamma_{opx}}(\phi_f, \phi_{opx}, c_f, p),\end{aligned} \tag{4.94}$$

where we introduce the notation $\boldsymbol{l}^k(z)$ to extract the boundary terms of the solution to compute the numerical flux. In particular, we have

$$\begin{aligned}\boldsymbol{l}^k(z_B^k) &= (1, 0, \ldots, 0)^T \\ \boldsymbol{l}^k(z_T^k) &= (0, 0, \ldots, 1)^T\,.\end{aligned} \tag{4.95}$$

A more preferable form of (4.94) is achieved by multiplying though by the inverse of

the mass matrix, yielding

$$\begin{aligned}\frac{d\boldsymbol{\phi_f}}{dt} =& -\frac{\gamma}{\delta}\mathcal{D}^{(z,k)}\boldsymbol{\phi_f} + \left(\mathcal{M}^{(z,k)}\right)^{-1}\left[\boldsymbol{l}^k(z)\left(\frac{\gamma}{\delta}\boldsymbol{\phi_f} - \left(\frac{\gamma}{\delta}\boldsymbol{\phi_f}\right)^*\right)\right]_{z_B^k}^{z_T^k} \\ &+ \boldsymbol{p}{\boldsymbol{\phi_f}}^m - \boldsymbol{\Gamma_{opx}}(\phi_f, \phi_{opx}, c_f, p)\end{aligned} \tag{4.96}$$

In Eq. (4.96) we have effectively discretized the spatial component of partial differential equation (4.38), and now simply must evolve the solution using a time integration scheme. We compute the numerical flux term $\left(\frac{\gamma}{\delta}\boldsymbol{\phi_f}\right)^*$ with a Lax-Friedrichs flux.

The evolution of ϕ_{opx} (Eq. (4.39)) is similar:

$$\begin{aligned}\frac{d\boldsymbol{\phi_{opx}}}{dt} =& -\frac{\gamma}{\delta}\mathcal{D}^{(z,k)}\boldsymbol{\phi_{opx}} + \left(\mathcal{M}^{(z,k)}\right)^{-1}\left[\boldsymbol{l}^k(z)\left(\frac{\gamma}{\delta}\boldsymbol{\phi_{opx}} - \left(\frac{\gamma}{\delta}\boldsymbol{\phi_{opx}}\right)^*\right)\right]_{z_B^k}^{z_T^k} \\ &+ \frac{\eta}{1-\beta}\boldsymbol{\Gamma_{opx}}(\phi_f, \phi_{opx}, c_f, p)\end{aligned} \tag{4.97}$$

Again, a Lax-Friedrichs flux is used. Evolving the fluid concentration requires a bit more effort. Denoting the (local) fluid velocity components

$$\mathbf{V}_f = (\boldsymbol{u}, \boldsymbol{w}), \tag{4.98}$$

we have

$$\begin{aligned}\frac{d\left(\boldsymbol{\phi_f c_f}\right)}{dt} =& -\delta^{-1}\left\{\mathcal{D}^{(z,k)}\left(\boldsymbol{\phi_f c_f w}\right) - \left(\mathcal{M}^{(z,k)}\right)^{-1}\left[\boldsymbol{l}^k(z)\left(\boldsymbol{\phi_f c_f w} - \left(\boldsymbol{\phi_f c_f w}\right)^*\right)\right]_{z_B^k}^{z_T^k}\right\} \\ &- \delta^{-1}\left(\boldsymbol{\phi_f c_f w}\right)\left(\mathcal{D}^x\right)^T \\ &- \delta^{-1}\boldsymbol{\Gamma_{opx}}(\phi_f, \phi_{opx}, c_f, p)\end{aligned} \tag{4.99}$$

For the numerical flux $\left(\boldsymbol{\phi_f c_f w}\right)^*$ we use a global Lax-Friedrichs flux. Numerical experiments have revealed that the highly nonlinear product $\boldsymbol{\phi_f c_f w}$ is at significant risk for aliasing errors. We handle this issue with the combination of filtering and higher polynomial order ($N \simeq 20$).

The only remaining component we need in order to evolve Eqs. (4.96),(4.97), and (4.99) is the compaction rate p. For this we need to solve the elliptic equation (4.41).

To accommodate for non-zero Dirichlet boundary conditions $p(0) = p_0$, (see Eq. (4.58)), we write

$$p = \frac{\tilde{p}}{\delta} + p_0 \tag{4.100}$$

so that $\tilde{p}(z = 0) = 0$, and rewrite (4.41) as

$$-\nabla \cdot \left[\phi_f^n \nabla \tilde{p}\right] + \tilde{p}\phi_f^m = -\delta \phi_f^m p_0 + \delta \frac{\partial}{\partial x}\left(\phi_f^n \frac{dp_0}{dx}\right) - \frac{\partial \phi_f^n}{\partial z} := f. \tag{4.101}$$

Using the previous selections of numerical flux (for the derivatives of ϕ_f), we are able to discretize the right hand side $f \simeq \boldsymbol{f}$. From here we write the scheme

$$\begin{aligned} -\mathcal{M}^{(z,k)} \left[\left(\boldsymbol{\phi_f}^n \left(\tilde{\boldsymbol{p}} \left(\mathcal{D}^x\right)^T\right)\right)\left(\mathcal{D}^x\right)^T + \mathcal{D}^{(z,k)}\left(\boldsymbol{\phi_f}^n \boldsymbol{q}\right) - \tilde{\boldsymbol{p}}\boldsymbol{\phi_f}^m\right] \\ + \left[\boldsymbol{l}^k(z)\left(\boldsymbol{q} - \boldsymbol{q^*}\right)\right]_{z_B^k}^{z_T^k} = \mathcal{M}^{(z,k)}\boldsymbol{f}, \\ \boldsymbol{q} = \mathcal{D}^{(z,k)}\boldsymbol{u} - \left(\mathcal{M}^{(z,k)}\right)^{-1}\left[\boldsymbol{l}^k(z)\left(\tilde{\boldsymbol{p}} - \tilde{\boldsymbol{p}}^*\right)\right]_{z_B^k}^{z_T^k}. \end{aligned} \tag{4.102}$$

For the numerical fluxes $\tilde{\boldsymbol{p}}^*$ and $\boldsymbol{q}^*$ we use a central flux and stabilized central flux, respectively. The former choice is made because p has no preferred direction of propagation; the latter is made in order to guarantee invertibility of the operator. These are given as

$$\tilde{\boldsymbol{p}}^* = \{\{\tilde{\boldsymbol{p}}\}\}, \quad \boldsymbol{q}^* = \{\{\boldsymbol{q}\}\} - \tau\left[\tilde{\boldsymbol{p}}\right] \tag{4.103}$$

where $\{\{\cdot\}\}$ represents the average of the quantity at the interface and $[\cdot]$ represents the jump. τ is a positive number, which we take to be 1. The role of this parameter is to penalize the solution to disallow jumps in $\tilde{\boldsymbol{p}}$. For a greater discussion of the details of the stabilized central flux, as well as other possible choices to use, see [3].

Assembling (4.102) over all k, the resulting linear system is

$$\mathcal{H}\tilde{\boldsymbol{p}} = \mathcal{M}\boldsymbol{f}, \tag{4.104}$$

where $\mathcal{M}$ is the global mass matrix, with $\tilde{\boldsymbol{p}}$ and $\boldsymbol{f}$ also representing global arrays. From this point we may utilize a standard linear systems solver. Due to the nonlinear ϕ_f^n permeability terms, $\mathcal{H}$ is not symmetric. In the work presented, a matrix-free implementation of the BiCGStab iterative method is used to solve the linear system. For a preconditioner, we compute the incomplete Cholesky factorization of the operator $\mathcal{H}$ with a constant coefficient ($\phi_f = 1$), and drop tolerance 10^{-6}.

4.4.5 Time Integration

For time integration, a fourth-order, explicit, low-storage Runge-Kutta (LSERK) method is implemented, with CFL number 0.5. As Da tends to infinity, the reaction term Γ_{opx} becomes stiff in the time step restriction, perhaps motivating an alternative formulation. However, it has been observed numerically that temporal accuracy is important in resolving the fast-reacting features present in large Da simulations, so to date the high-order Runge-Kutta scheme has been necessary for these cases. The time step restriction in these large (but finite) Da cases motivates us to consider the equilibrium ($Da = \infty$) setting, which is much easier to integrate in time due to the removal of the Da factor in the reaction term.

4.4.6 Additional Remarks

For help in stabilizing the nonlinear terms (particularly those in the fluid velocity $\mathbf{V}_f$), weak artificial dissipation is introduced to the scheme via an exponential filter in each direction. The relatively weak strength choice $s = 16$ is used for each direction, with cutoff frequencies of 0 and 6 in the horizontal and vertical directions, respectively. The solution variables ϕ_f and p are filtered within each Runge-Kutta

stage.

The term

$$\left(\mathcal{M}^{(z,k)}\right)^{-1}\left[\boldsymbol{l}^k(z)\right]_{z_B^k}^{z_T^k}$$

appears often in the numerical flux computations. It is useful to define an operator which evaluates this expression.

Lastly, it is noted that the opx concentration ϕ_{opx} is not allowed to attain non-physical, negative values. In the kinetic formulation ($Da < \infty$) this would, asymptotically, not occur anyway, as the surface area term in Γ_{opx} eliminates dissolution for near-zero values of ϕ_{opx}. In the equilibrium formulation, however, there is no such surface area term and so we set $\Gamma_{opx} = 0$ when ϕ_{opx} becomes negative, while fixing the opx concentration to 0. In future work we will explore the implementation of a level set to more precisely treat the $\phi_{opx} = 0$ region, where dissolution must be turned off.

Chapter 5

Validation of Numerical Model

Having presented a suitable model for analysis, we next set our sights on validation of the model. In this chapter we detail a linear stability analysis, demonstrate convergence of the kinetic to the equilibrium formulation and agreement in the growth rates from the transient solutions with those predicted from the linear stability analysis.

5.1 Linear Stability Analysis for Kinetic Formulation

We perform a linear stability analysis on the equations (4.38)-(4.43). This study will both help us to better understand the behavior of the system as well as providing a key benchmark that may be used for validation of the transient solution. For each unknown u, $u \in \{\phi_f, \phi_{opx}, c_f, p\}$, we write

$$u(x, z, t) = \bar{u} + \epsilon \tilde{u} e^{ikx+\sigma t}. \tag{5.1}$$

The base state $\bar{u}$ is assumed to be a function of z only, $\epsilon \ll 1$ is a small number, and $k \in [0, \infty)$ is the (horizontal) perturbation frequency. σ is the growth rate. With these assumptions we may substitute into the simulation equations (4.38)-(4.43).

Keeping only $O(\epsilon)$ terms we arrive at the system

$$\begin{pmatrix} A_{S\phi_f} & A_{S\phi_{opx}} & A_{Sc_f} & A_{Sp} \\ 0 & A_{O\phi_{opx}} & A_{Oc_f} & 0 \\ A_{C\phi_f} & A_{C\phi_{opx}} & A_{Cc_f} & A_{Cp} \\ A_{P\phi_f} & 0 & 0 & A_{Pp} \end{pmatrix} \begin{pmatrix} \tilde{\phi}_f \\ \tilde{\phi}_{opx} \\ \tilde{c}_f \\ \tilde{p} \end{pmatrix} = \sigma \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & \delta\bar{\phi}_f & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} \tilde{\phi}_f \\ \tilde{\phi}_{opx} \\ \tilde{c}_f \\ \tilde{p} \end{pmatrix} \tag{5.2}$$

where the entries are given as

$$\begin{aligned}
A_{S\phi_f} &= -\tfrac{\gamma}{\delta}\mathcal{D} + m\bar{\phi}_f^{m-1}\bar{p} \\
A_{S\phi_{opx}} &= Dal\bar{\phi}_{opx}^{l-1}\left(z-\bar{c}_f\right) \\
A_{Sc_f} &= -Da\bar{\phi}_{opx}^{l} \\
A_{Sp} &= \bar{\phi}_f^m \\
A_{O\phi_{opx}} &= -\tfrac{\gamma}{\delta}\mathcal{D} - Dal\bar{\phi}_{opx}^{l-1}\tfrac{\eta}{1-\beta}\left(z-\bar{c}_f\right) \\
A_{Oc_f} &= Da\tfrac{\eta}{1-\beta}\bar{\phi}_{opx}^{l} \\
A_{C\phi_f} &= \bar{c}_f'\left[n\bar{\phi}_f^{n-1}\left(\delta\bar{p}'-1\right)-\gamma\right] \\
A_{C\phi_{opx}} &= Dal\bar{\phi}_{opx}^{l-1}\left(z-\bar{c}_f\right)\left(1-\delta\bar{c}_f\right) \\
A_{Cc_f} &= \left[\bar{\phi}_f^n\left(\delta\bar{p}'-1\right)-\gamma\bar{\phi}_f\right]\mathcal{D} - Da\bar{\phi}_{opx}^{l}\left(1+\delta\left(z-2\bar{c}_f\right)\right) \\
A_{Cp} &= \delta\bar{c}_f'\bar{\phi}_f^n\mathcal{D} \\
A_{P\phi_f} &= \delta m\bar{\phi}_f^{m-1}\bar{p} - n\left[\left(\delta\bar{p}'-1\right)\left(\bar{\phi}_f^{n-1}\mathcal{D}+\left(\bar{\phi}_f^{n-1}\right)'\right)-\delta\bar{\phi}_f^{n-1}\bar{p}''\right] \\
A_{Pp} &= \delta\left[\bar{\phi}_f^m - \mathcal{D}n\bar{\phi}_f'\bar{\phi}_f^{n-1} + \left(k^2-\mathcal{D}^2\right)\bar{\phi}_f^n\right].
\end{aligned} \tag{5.3}$$

As $\tilde{p}$ does not have a growth rate associated with it (p is determined by ϕ_f), the 4-by-4 block eigenvalue problem (5.2) may be simplified by first writing

$$\tilde{p} = -\left(A_{Pp}\right)^{-1}A_{P\phi_f}\tilde{\phi}_f \tag{5.4}$$

and then substituting the first and third rows' $\tilde{p}$-component accordingly. The resulting 3-by-3 block is then solved easily with a standard eigenvalue solver.

The base states $\left(\bar{\phi}_f, \bar{\phi}_{opx}, \bar{c}_f, \bar{p}\right)$ taken as the leading terms of the one-dimensional

steady states (4.53)-(4.56),

$$\bar{\phi}_f = \left(1 + q_0 \frac{\delta z}{1-\delta z}\right)^{1/n} \tag{5.5}$$

$$\bar{\phi}_{opx} = 1 - \frac{\eta q_0}{\gamma\,(1-\beta)} \frac{\delta z}{1-\delta z} \tag{5.6}$$

$$\bar{c}_f = z - \frac{q_0}{Da - \delta q_0} \tag{5.7}$$

$$\bar{p} = -\frac{q_0}{(1-\delta z)^2} \tag{5.8}$$

5.2 Linear Stability Analysis for Equilibrium Formulation

In the event that $Da = \infty$, the concentration is fixed at equilibrium ($c_f = z$) and the reaction rate has no dependence on ϕ_{opx}. Therefore ϕ_f (and in turn, p) are decoupled from ϕ_{opx} in the stability analysis. Accordingly, the linear system (5.2)-(5.3) simplifies to a 2-by-2 block,

$$\begin{pmatrix} A^{\infty}_{S\phi_f} & A^{\infty}_{Sp} \\ A_{P\phi_f} & A_{Pp} \end{pmatrix} \begin{pmatrix} \tilde{\phi}_f \\ \tilde{p} \end{pmatrix} = \sigma \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} \tilde{\phi}_f \\ \tilde{p} \end{pmatrix} \tag{5.9}$$

with modified fluid concentration entries

$$\begin{aligned} A^{\infty}_{S\phi_f} &= -\tfrac{\gamma}{\delta}\mathcal{D} + m\bar{\phi}_f^{m-1}\bar{p} - \tfrac{\gamma - n\bar{\phi}_f^{n-1}(\delta\bar{p}'-1)}{\delta z - 1} \\ A^{\infty}_{Sp} &= \bar{\phi}_f^{m} - \tfrac{\delta\bar{\phi}_f^{n}}{1-\delta z}\mathcal{D}. \end{aligned} \tag{5.10}$$

The entries $A_{P\phi_f}, A_{Pp}$ for the pressure equation remain the same as the kinetic case. In the same fashion as before, we may reduce (5.9)-(5.10) to a single block by replacing $\tilde{p}$ as a linear combination of $\tilde{\phi}_f$.

5.3 Convergence of Kinetic to Equilibrium Formulations

One measure of validation we may conduct through the linear stability analysis (5.2)-(5.10) is to demonstrate convergence of the kinetic to equilibrium formulations in the limit $Da \to \infty$. This convergence is crucial, as it may justify the usage of the latter formulation for transient numerical simulations. We disregard dependence of the porosity in the viscosity ($m = 0$) and remove the opx surface area term from reaction ($l = 0$).

Tests are performed with an upwelling rate $\gamma = 0.00025$ and increasing Da, presented in Figure 5.1. The top left panel of this figure shows a close match between the growth rates in kinetic and equilibrium formulations for increasing Da. In the top right we measure the convergence of the eigenfunctions $\tilde{\phi}_f$ and $\tilde{p}$, and the eigenvalue σ for a fixed $k = 100$. That is, we measure the relative differences

$$\frac{\left|\tilde{\phi}_f^{Da} - \tilde{\phi}_f^{\infty}\right|_h}{\left|\tilde{\phi}_f^{\infty}\right|_h}, \frac{\left|\tilde{p}^{Da} - \tilde{p}^{\infty}\right|_h}{\left|\tilde{p}^{\infty}\right|_h}, \qquad \left|\frac{\sigma^{Da} - \sigma^{\infty}}{\sigma^{\infty}}\right|$$

where $|\cdot|_h$ is the discrete L^2-norm and quantities with a superscript Da refer to kinetic results while those with superscript ∞ refer to the equilibrium. From these results we see a clear convergence of eigenfunctions and eigenvalues between the kinetic and equilibrium cases. For reference, in the bottom left and right panels of Figure 5.1 we plot the eigenfunctions $\tilde{\phi}_f, \tilde{p}$ for the entire spectrum $0 \leq k \leq 240$ in increments of 20. The eigenfunctions have been normalized so that $\tilde{\phi}_f(z_H) = \tilde{p}(z_H) = 1$.

5.4 Matching Transient Growth Rate to Linear Stability Analysis

The next validation we perform is to check that the growth of the transient solutions to Eqs. (4.38)-(4.43) match up to the predicted growth rates that were computed in Figure 5.1. In Figure 5.2 these results are presented for the equilibrium formulation. Transient growth rates are approximated by measuring the rate of change of the Fourier coefficient corresponding to the k^{th} mode. We observe a close matching of the transient simulations to the predicted linear growth rates, as shown in the right panel of this figure.

One may notice in the transient simulation growth rate results presented in Figure 5.2 that the time scale for equilibrium (top right) and kinetic formulations (bottom right) are wildly different: the former is of order 1 while the latter is on the order of 10^{-7}. This is due to the kinetic formulation having a time step restriction $\Delta t \sim Da^{-1}$. For the results given in Figure 5.2, we have taken $Da = 10^6$. The equilibrium formulation does not carry this restriction.

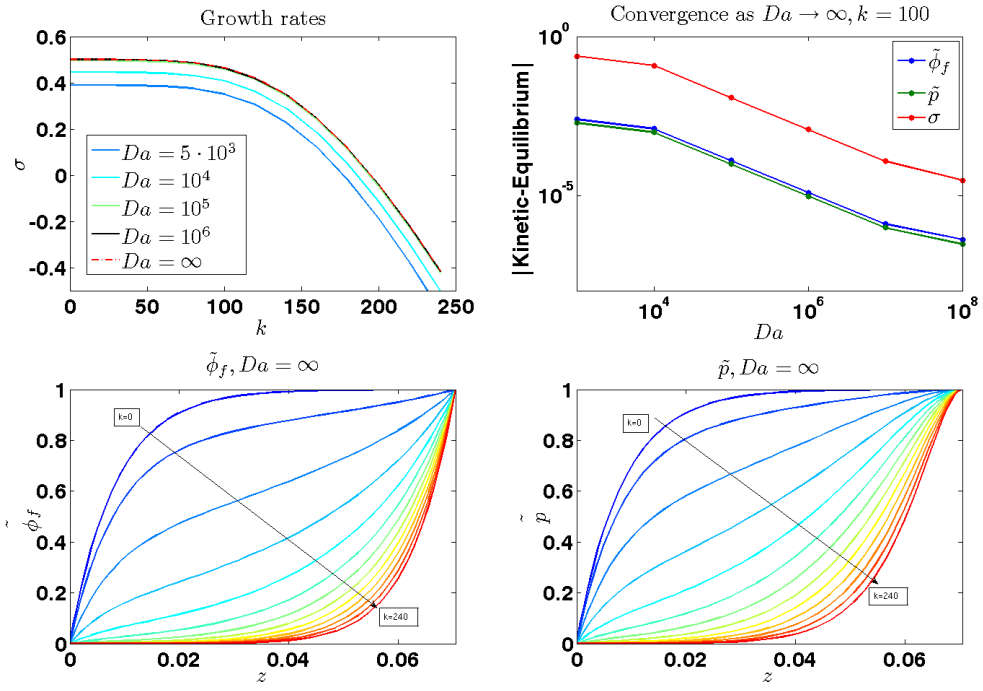

Figure 5.1: Linear stability analysis, as computed in Eqs. (5.2)-(5.10). $\gamma = 0.00025$. Top Left: computed growth rates σ vs. perturbation frequency k, for varying Da. Top Right: convergence in $\tilde{\phi}_f, \tilde{p}, \sigma$ as $Da \to \infty$. A fixed perturbation frequency of $k = 100$ is chosen. Bottom left/right: $\tilde{\phi}_f/\tilde{p}$ eigenfunctions for $k \in [0, 240]$, plotted in increments of 20. All eigenfunctions have been normalized so that $\tilde{\phi}_f(z_H) = \tilde{p}(z_H) = 1$.

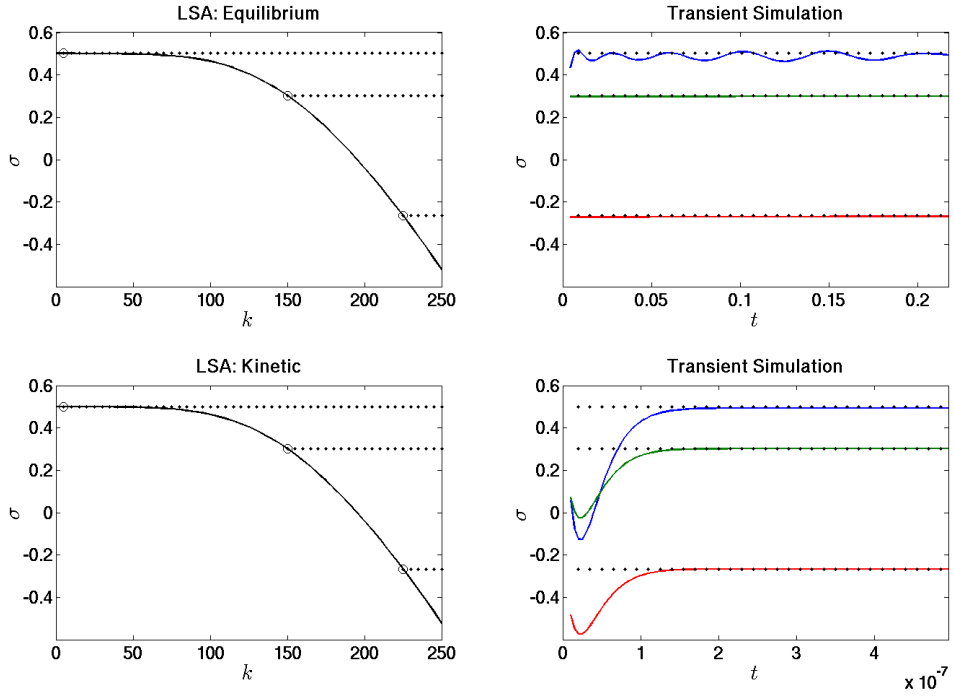

Figure 5.2: Comparison of predicted growth rates σ from linear stability analysis (LSA) from Eqs. (5.2)-(5.10) to those actually measured from the transient simulations (4.38)-(4.43). Three perturbation frequencies, $k = 5, 150, 225$ are examined for the transient simulations. Growth rates in the transient simulations are computed by considering the rate of change of the magnitude of the Fourier coefficient corresponding to frequency k. Top: Equilibrium formulation. Bottom: Kinetic formulation ($Da = 10^6$) with surface area coefficient $l = 0.5$. For both cases we take $m = 0$.

Chapter 6

Observations from Numerical Model

Having defined the model in Chapter 4 and validated it in Chapter 5, we next turn our attention to exploring features of the model. In this chapter we will first discuss impacts on the stability of the l and m parameters, which control the influence of the opx mineral in reaction and the porosity in bulk viscosity, respectively. Next, we introduce a different non-dimensionalization based on the upwelling velocity. Numerical simulations from this dimensionalization demonstrate a clear link between mantle heterogeneity, represented by inflow ϕ_f perturbations, and the formation of melt channels.

6.1 Effects of the ϕ_f^m Viscosity Term

Results from the linear stability analysis above demonstrate convergence in the limit $Da \to \infty$ of the kinetic to the equilibrium formulations. Next, we consider the same problem but with varying dependence of the porosity ($\phi_f^m, m \in [0, 1]$) in the viscosity term.

In Figure 6.1 the predicted growth rates from the linear stability analysis are given

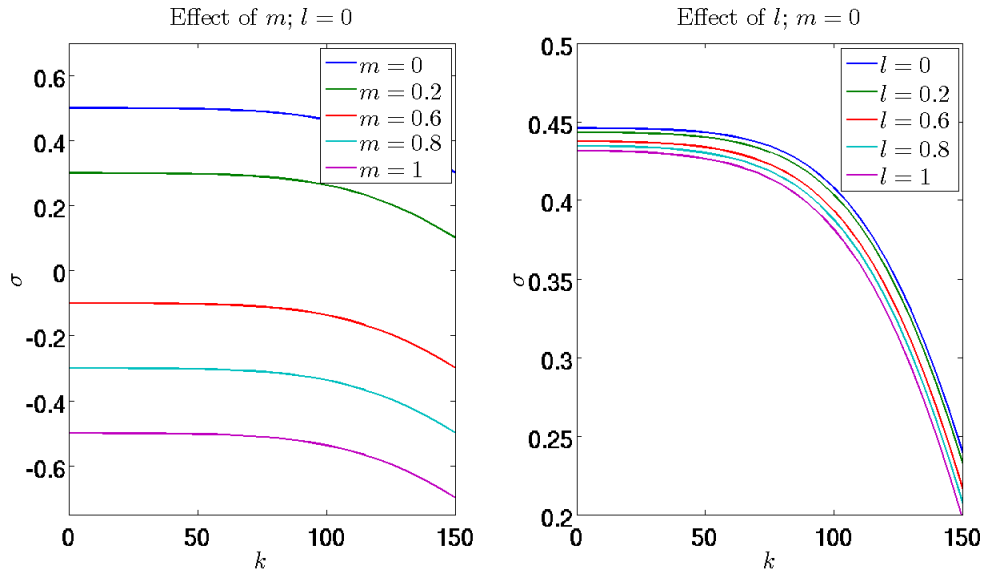

Figure 6.1: Growth rates σ vs. horizontal perturbation frequency k from linear stability analysis. $\gamma = 0.00025$. Left: fixing $l = 0$, we vary the porosity component ϕ_f^m in bulk viscosity. A reaction rate constant of $Da = 10^6$ is used, which is essentially converged to the equilibrium case. Right: fixing $m = 0$, we vary the surface area component ϕ_{opx}^l in reaction rate. A reaction rate constant of $Da = 10^4$ is used. Taking both $l, m \to 1$ stabilizes the system.

with varying m, and fixed $\gamma = 0.00025$, $Da = 10^6$, and $l = 0$, the exponent in the opx mineral in the surface term. It is observed that the growth rate σ varies linearly with the m exponent, so that including the presence of porosity in the viscosity term acts to stabilize the system.

6.2 Effects of the ϕ_{opx}^l Surface Area Term

In the transient numerical simulations presented in Sec. 6.4 it was discovered that the presence of a ϕ_{opx}-dependent surface area term in the kinetic formulation was crucial in acting to turn off reaction once the mineral opx had been dissolved. For the equilibrium formulation this term was no longer present, so that in order to match up the transient solutions we had to include a condition to manually turn off reaction once opx is exhausted (Eqs. (6.19) and (6.20)).

Extrapolating these results to our rescaled equations, it is significant to observe that the convergence of kinetic to equilibrium formulations in the linear stability analysis, shown in Figure 5.1, occurs with opx exponent $l = 0$, i.e., no inclusion of the ϕ_{opx} surface area term in the reaction rate. In the right panel of Figure 6.1 we plot the growth rate for varying $l \in [0, 1]$ and reaction rate constant $Da = 10^4$. As with the ϕ_f term in the viscosity, the ϕ_{opx}^l surface area term stabilizes the equation, with increasing effectiveness as $l \to 1$. This feature is quite minimal, however, and for sufficiently high Da the dispersion relationships for varying l are nearly indistinguishable.

6.3 Dimensionless Governing Equations on an Upwelling Time Scale

Previous numerical results have been given with a reactive time scale, specified in Eq. (4.31). In this section we present a slightly different non-dimensionalization, where the time scale is normalized to the upwelling velocity. The motivation behind this dimensionalization is to explore the effects of a source perturbation, simulating mantle heterogeneity, on the sustained formation of melt channels. Transient numerical results will follow in subsequent sections. The surface area and viscosity exponents (l and m, respectively) have been fixed at 1.

We now assume length, pressure and time scales as

$$\begin{aligned} L &= \sqrt{\tfrac{\kappa_0 \xi_0}{\mu \phi_f^0}} \\ P &= \tfrac{\xi_0}{T} \\ T &= \tfrac{L}{W_0} \\ \kappa_0 &= \tfrac{\left(\phi_f^0\right)^n d^2}{b} \end{aligned} \tag{6.1}$$

δ	β	η_0	ϕ_f^0	Pe	Da	R
10^{-2}	0.5	0.133	0.02	∞	$10^2 \to 10^5, \infty$	100

Table 6.1: Typical values for the dimensionless parameters given in (6.2).

The system is described by the following dimensionless parameters

$$\begin{gathered} \phi_f^0, \qquad \eta_0 = \tfrac{\phi_f^0}{\phi_{opx}^0}, \qquad \delta = (1-\beta)\,\tfrac{\Delta c}{\hat{c}}, \\ Pe_s = \tfrac{W_0 L}{D}, \quad Da_s = DaR = \tfrac{\alpha \hat{c} L \phi_{opx}^0}{W_0}, \qquad R = \tfrac{\kappa_0 g \Delta\rho}{\mu \phi_f^0 W_0} \end{gathered} \tag{6.2}$$

where $\hat{c} = c_{opx}^0 - \beta c_{ol}^0 - (1-\beta)\, c_f^0$. The parameter δ effectively controls the slope of the solubility gradient ($f(z)$ in (4.25)), with stoichiometric dissolution constant β. The reaction rate constant is given by the product $Da_s = DaR$, where Da is a measure of the dissolution rate relative to the rate of melt flow in the porous rock and is referred to as the Damköhler number. The Peclet number Pe_s is a measure of the relative rate of solid upwelling to the rate of diffusion in the melt. In these simulations, we neglect diffusion and so take $Pe_s = \infty$. Finally, the parameter R is a measure of the ratio of the fluid velocity to solid upwelling velocity. Therefore, higher R corresponds to less upwelling. Typical values which we choose for these parameters are given in Table 6.1.

Dropping the $*$ superscripts on dimensionless variables, from these simplifications we arrive at the following dimensionless set of equations for $\phi_f, \phi_{opx}, c_f, p$, and $\mathbf{V}_f$:

$$\frac{\partial \phi_f}{\partial t} = -\phi_f^0 {\phi_f}^2 p - \frac{\partial \phi_f}{\partial z} - \frac{\Delta V}{W_0} \tilde{\mathbf{V}}_s \cdot \nabla \phi_f + \phi_f p - \delta \Gamma_{opx} \tag{6.3}$$

$$\frac{\partial \phi_{opx}}{\partial t} = -\frac{\partial \phi_{opx}}{\partial z} - \frac{\Delta V}{W_0} \tilde{\mathbf{V}}_s \cdot \nabla \phi_{opx} - \phi_f^0 \phi_f \phi_{opx} p + \frac{\eta_0 \delta}{1-\beta} \Gamma_{opx} \tag{6.4}$$

$$\frac{\partial \phi_f c_f}{\partial t} = -R \nabla \cdot \left[c_f \phi_f \mathbf{V}_f - \frac{\phi_f}{Pe_s} \nabla c_f \right] - \Gamma_{opx} \tag{6.5}$$

$$-\nabla \cdot \left[\phi_f^n \nabla p \right] + \phi_f p = -R \frac{\partial}{\partial z} \left[\phi_f^n \left(1 - \phi_f^0 \phi_f \right) \right] \tag{6.6}$$

$$\phi_f \mathbf{V}_f = \phi_f \mathbf{n}_z + \frac{\Delta V \tilde{\mathbf{V}}_s}{W_0} - \phi_f^n \left(\nabla p - R \left(1 - \phi_f^0 \phi_f \right) \mathbf{n}_z \right) \tag{6.7}$$

To further simplify matters we disregard terms of order ϕ_f^0 and ΔV. Doing so yields

$$\frac{\partial \phi_f}{\partial t} = -\frac{\partial \phi_f}{\partial z} + \phi_f p - \delta \Gamma_{opx} \tag{6.8}$$

$$\frac{\partial \phi_{opx}}{\partial t} = -\frac{\partial \phi_{opx}}{\partial z} + \frac{\eta_0 \delta}{1-\beta} \Gamma_{opx} \tag{6.9}$$

$$\frac{\partial \phi_f c_f}{\partial t} = -R \nabla \cdot \left[c_f \phi_f \mathbf{V}_f - \frac{\phi_f}{Pe_s} \nabla c_f \right] - \Gamma_{opx} \tag{6.10}$$

$$-\nabla \cdot \left[\phi_f^n \nabla p \right] + \phi_f p = -R \frac{\partial \phi_f^n}{\partial z} \tag{6.11}$$

$$\phi_f \mathbf{V}_f = \phi_f \mathbf{n}_z - \phi_f^n (\nabla p - R \mathbf{n}_z) \tag{6.12}$$

The reaction term is

$$\Gamma_{opx} = \begin{cases} -Da_s \phi_f \phi_{opx} \left(z - c_f \right), Da_s < \infty \\ \frac{1}{1-\delta z} \left(\frac{1}{Pe_s} \frac{\partial \phi_f}{\partial z} - R \phi_f w \right), Da_s = \infty \end{cases} \tag{6.13}$$

The porosity ϕ_f has now been included in the surface area term in reaction; this component does not significantly affect numerical results. The equations (6.8) - (6.13) form the system that we will explore in depth with numerical simulation. As in the previous non-dimensionalization, we modify the c_f and p boundary conditions to remove the artificial compaction layer near inflow. These are given as

$$c_f(0) = \frac{1+R}{\delta \left(1 + R \right) - Da_s} \tag{6.14}$$

$$p(0) = \begin{cases} \delta \left(\frac{1+R}{1+nR} + Da_s c_f(0) \right), & Da_s < \infty \\ \delta \left(\frac{1+R}{1+nR} - 1 - R \right), & Da_s = \infty \end{cases} \tag{6.15}$$

As before, $\phi_{opx}(0) = 1$ and $\phi_f(0) = 1 + h(x,t)$, for a small, prescribed perturbation h.

The maximum domain length of interest z_H (corresponding to the approximate location of $\phi_{opx} = 0$ in the one-dimensional steady state) is

$$z_H = \delta^{-1} \frac{1-\beta}{(1+R)\,\eta + 1 - \beta} \tag{6.16}$$

6.4 Transient Numerical Results for the Kinetic Formulation

In this section we present numerical simulations on the dimensionless equations (6.8)-(6.13). A strong correlation between mantle heterogeneity, represented by a perturbation in ϕ_f at inlet, and the formation of melt channels is discovered. We investigate this and other behavior for the kinetic formulation. In the subsequent section, results are linked with the equilibrium formulation.

6.4.1 Correlation of Inflow ϕ_f- Perturbations to Outflow Melt Channels

One remarkable feature observed is a strong correlation of the mantle heterogeneity (represented as perturbations in the lower ϕ_f boundary condition) to the formation of channels higher in the domain. To demonstrate this we consider a transient simulation with the following prescribed boundary conditions:

$$\phi_f(x, z=0, t) = \begin{cases} 1 + \text{Noise}(5), & 0 \le t \le 7.33 \\ 1 + \text{Noise}(10), & 7.33 < t \le 11 \\ 1 + 0.05 \exp\left[-100\left((x-1/6)^2 + S(t)\,(x-1/2)^2 + (x-5/6)^2\right)\right], & 11 < t \le 20 \end{cases} \tag{6.17}$$

where Noise(k) is Gaussian white noise in the first k Fourier frequencies with variance 10^{-3}, and the switch function $S(t)$ returns 0 if the integer part of t is odd, and 1 otherwise.

In Figure 6.2, results are shown. In the bottom panel, the perturbed incoming ϕ_f, corresponding to (6.17), is shown, while in the top panel, the outgoing ϕ_f is given. We may observe a direct correlation of the two, although it is off by a few (roughly 2) dimensionless time units. This is due to the time it takes for heterogeneities at the base to rise to the top of the simulation domain.

Initially, the smoother perturbation of the system ($0 \leq t \leq 7.33$) leads to well-defined, but time-varying channels at outflow. For the higher frequency noise on the time interval $7.33 < t < 11$, narrower, more defined channels are observed at outflow. Finally, the alternating 2-3 Gaussian bumps, prescribed on the time interval $11 < t \leq 20$, are observed directly on the outflow. As the middle bump disappears, the middle melt channel at outflow weakens but does not altogether disappear.

To better understand the transient behavior of this system with source perturbation as prescribed in Eq. (6.17), we plot the solution variables ϕ_f, ϕ_{opx}, p at the early time $t = 2.8$, which corresponds to the first perturbation in Eq. (6.17). For this study, however, we consider the much faster reacting rate of $Da = 1000$. These plots are given in Figure 6.3. For the solution time chosen, the ϕ_f inflow perturbation contains Gaussian white noise in the first five Fourier modes. It is observed that there are five channels appearing in the top of the simulation box, confirming the previous observations of relation between channel formation to source perturbation. Although the channel inflow has porosity around one, at the outflow the channels of melt have formed of porosity nearly 1.4, with accompanying compaction regions where the porosity is less than 0.5. The middle picture shows the opx fraction remaining, which corresponds as expected with the porosity (less fluid, more solid, and vice versa). The rightmost picture shows the compaction rate.

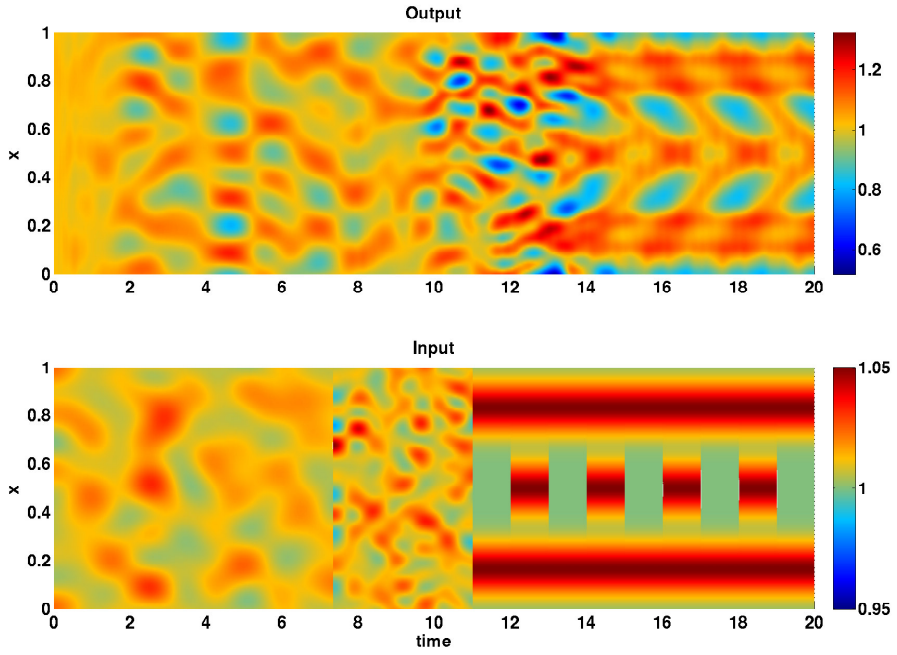

Figure 6.2: Comparison of input (bottom) perturbation to output (top) channels formed. Porosity ϕ_f is shown for a reaction rate of $Da = 10$ up to a dimensionless time of $t = 20$. Porosity perturbations from Eq. (6.17) are used: For the first 7.33 time units, a low-frequency perturbation with noise in the first five Fourier modes is used. For $7.33 < t < 11$, a higher frequency perturbation, with noise in the first 10 Fourier modes, is given. For $11 < t < 20$, a regulated series of three Gaussian bumps is prescribed, with the middle disappearing every other time unit.

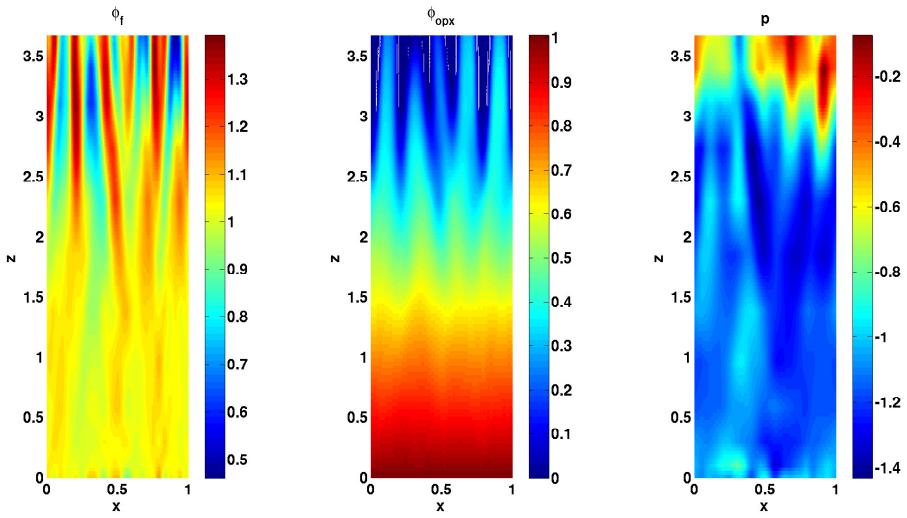

Figure 6.3: Transient simulation with perturbed ϕ_f inflow as pictured in the bottom panel of Figure 6.2. For this case, however, we take $Da = 1000$. The solution is examined at time $t = 2.8$, which corresponds to the perturbation in the first time regime in Eq. (6.17). Five channels of melt are observed, corresponding to the perturbation noise at inflow contained with the first five Fourier modes. Melt channels in ϕ_f correspond one-to-one with dissolved regions of ϕ_{opx}.

6.4.2 Channels Deepen as $Da \to \infty$

An important verification for the numerical solution is the effects of the reaction constant $Da_s = DaR$, where Da is the Damköhler number and the constant R effectively is the ratio of the fluid velocity to the solid (upwelling) velocity, which we fix at $R = 10^2$. Increasing Da causes faster dissolution of the mineral opx, which should lead to deeper melt channels.

Recalling the transient simulation with varying perturbation presented in Figures 6.2 and 6.3, we now instead consider the effects of increasing Da. From Figure 6.3 it is observed that the most interesting behavior occurs near the top of the simulation box, where melt channels have formed. Next, we consider the solution at outflow at the same time as in Figure 6.3, but with varying Da. These results are presented in Figure 6.4. We observe that for higher Da, there exist deeper channels of melt and greater dissolution of opx. For the most highly reacting cases of $Da = 500, 1000$, we

observe small Gibbs oscillations in the opx fraction. These may possibly be induced by the steep gradients (in the x direction) of ϕ_{opx}, and therefore, Γ_{opx}. Moreover, in the vertical direction, our choice of fixing ϕ_{opx} to 0 in the event that it go slightly below this threshold introduces a slight kink in the solution, leading to oscillations when evaluating the upwelling derivative.

The results given in Figures 6.2, 6.3, and 6.4 present a convincing case as to the correlation between channel formation and mantle heterogeneity, as well as the effects of increasing the Damköhler number leading to more pronounced channel formations. This latter observation is important because we mainly are interested in the case of $Da \to \infty$, with the hypothesis that high-permeability melt regions may act as conduits for melt migration to the surface. To better analyze this hypothesis, we next will consider the scenario of a fixed perturbation in ϕ_f at inflow, yielding a single melt channel.

6.4.3 Observations From a Single Melt Channel

A single channel of melt can be induced with the constant ϕ_f boundary perturbation

$$\phi_f\left(x, z=0, t\right) = 1 + 0.1 \exp\left[-100\left(x - .5\right)^2\right]. \tag{6.18}$$

With a constant perturbation, the system will ultimately reach a steady state. For transient numerical simulations, this state is typically reached at a dimensionless time in the neighborhood of the length of the z domain. In Figure 6.5, a transient solution for $Da = 500$ is given at dimensionless time $t = 2$. Here, the propagation of the boundary perturbation is observed, with compaction channels forming on the outside of the center channel of melt. A domain $(x, z) \in [0, 1] \times [0, 2]$ is used.

A few interesting features are observed near the top of the computational domain. As the opx is exhausted, the reaction rate tends to 0 as there is no mineral left to dissolve. This in turn contributes to a slight drop-off in the porosity, from which

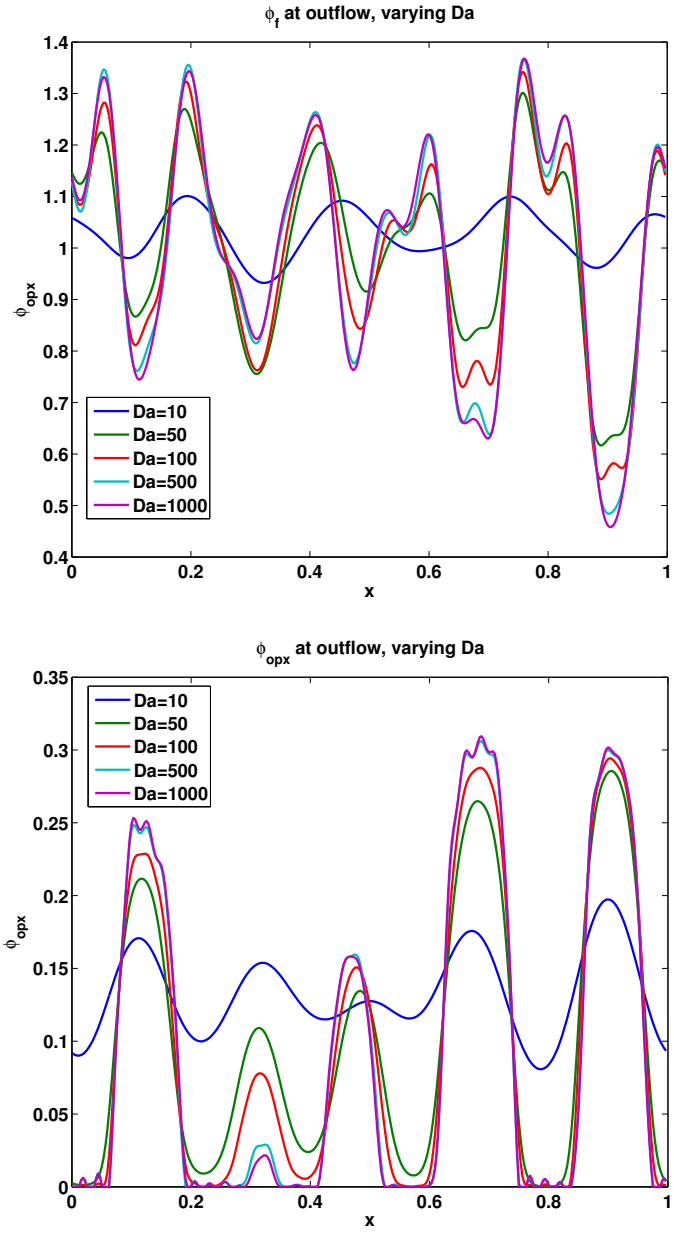

Figure 6.4: Solution ϕ_f and ϕ_{opx} at outflow ($z = 3.66$) for varying Da. The same time $t = 2.8$ as in Figure 6.3 is chosen, with perturbation defined in as in Eq. (6.17) and plotted in Figure 6.2. Higher Da produces deeper channels of melt and opx dissolution.

the compaction rate also rises. In the middle and lower panels of Figure 6.5 we plot focus on vertical profile centered at the channel midpoint $x = 0.5$ to better visualize this behavior.

As the profiles in Figure 6.5 demonstrate a significant change in the solution behavior when ϕ_{opx} tends to 0, it begs the question of what happens further out (i.e. higher in the z domain). A very interesting regime arises in this setting, as is seen in Figure 6.6. Here, we observe that the melt channel centered at $x = 0.5$ actually bifurcates in a "Y"-shape form soon after opx exhaustion. Moreover, there appear to be secondary channels forming near the top of the z domain, in the vicinity of $x = 0.25$ and $x = 0.75$.

We believe this behavior can be explained by the relationship between the opx fraction ϕ_{opx} and reaction rate Γ_{opx}. As in the shorter z domain case presented in Figure 6.5, the depletion of opx reduces the reaction rate, diminishing the fluid production and therefore ϕ_f as well. This occurs around $z = 2$. As we move upwards then, the regions of highest porosity are only able to sustain themselves by feeding on the opx that remains outside the middle of the now-dissolved channel. Therefore, the original melt channel splits for these regions. Meanwhile, the compaction regions (areas of low fluid fraction) on the outside of the original channel induce secondary channels as fluid from the lower half of the domain flows outside and around these compaction regions. To further understand the melt channel bifurcations, we present in Figure 6.7 the fluid velocity $\mathbf{V}_f = (v_x, v_z)$ fields and corresponding streamlines.

6.5 Transient Numerical Results for the Equilibrium Formulation

The results presented thus far have been for the kinetic formulation, i.e., finite reaction constant Da. The most physically relevant (and interesting, in lieu of the results given) setting is in the limit $Da \to \infty$. For large but finite Da, however, the

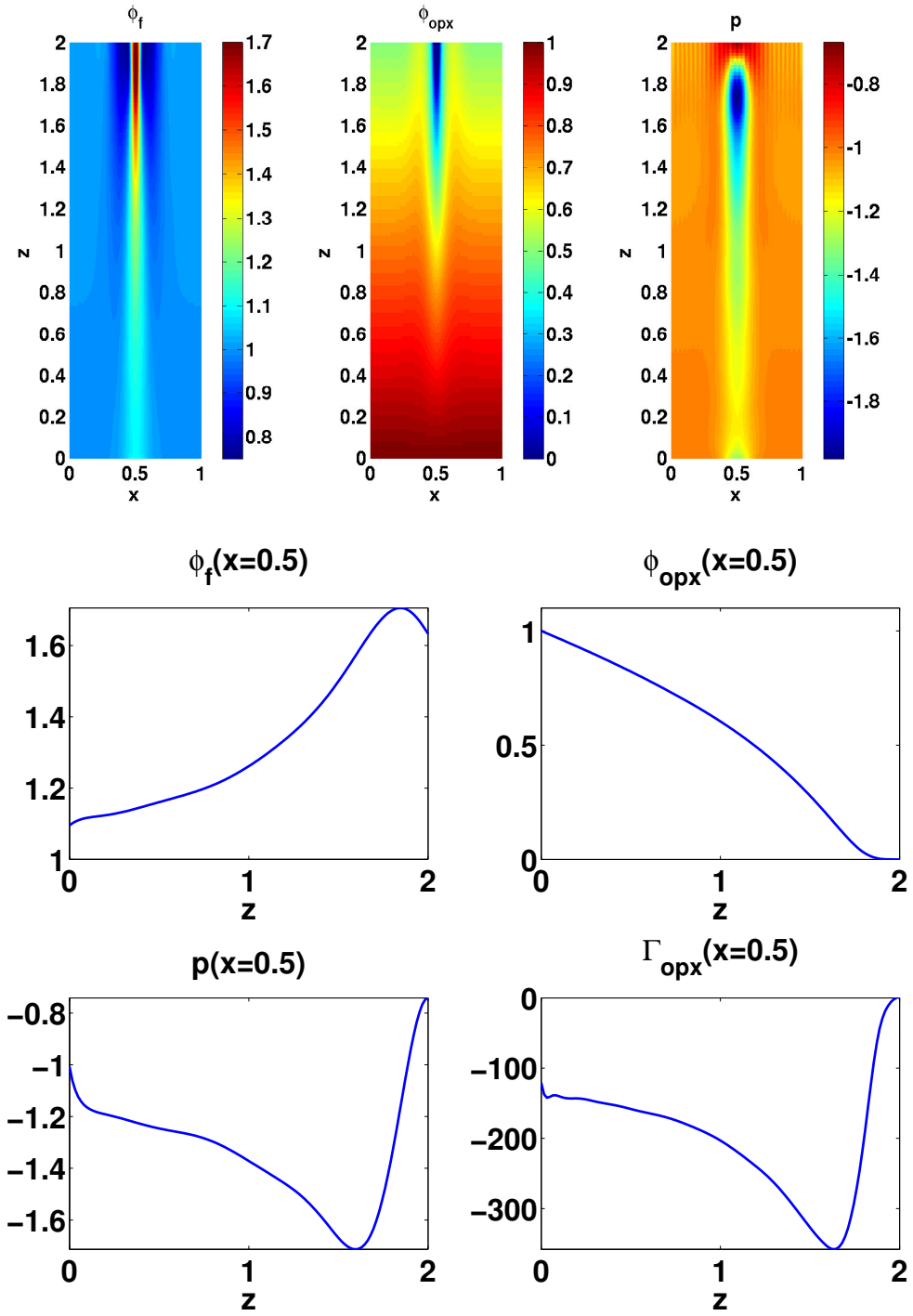

Figure 6.5: Transient solution with fixed boundary perturbation given in Eq. (6.18). Terminal time is $t = 2$. Top row: Two-dimensional profiles for $Da = 500$. Middle and bottom rows: solutions at in the channel midpoint $x = 0.5$ as a function of z. Once the opx mineral is exhausted, the porosity drops and compaction rate rises sharply.

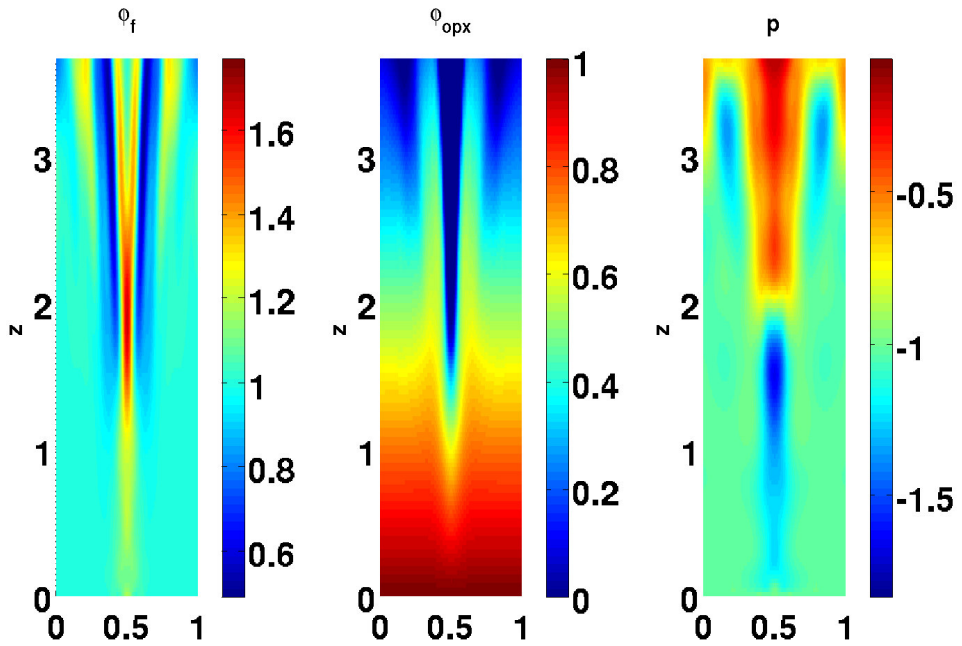

Figure 6.6: Transient solution with fixed boundary perturbation given in Eq. (6.18). Terminal time is $t = 2$. Reaction constant is $Da = 500$. Soon after the opx mineral is exhausted, the melt channel bifurcates into a "Y" formation.

time step in the explicit Runge-Kutta time integration will scale as $\Delta t \sim (Da)^{-1}$. To date, implementing alternative time stepping schemes (such as Strang splitting with an implicit method for the reaction terms) have not been fruitful, as temporal accuracy appears crucial in the Γ_{opx} term. However, we recall from earlier that it is possible to formulate an equilibrium ($Da = \infty$) version of the equations, with fixed $c_f(x, z, t) = z$. The reaction term for this case is

$$\Gamma_{opx} = \frac{1}{1-\delta z}\left(\frac{1}{Pe_s}\frac{\partial \phi_f}{\partial z} - R\phi_f w\right), \quad Da = \infty, \tag{6.19}$$

which no longer carries a stiff time step restriction. Here, $w = \mathbf{V}_f \cdot \mathbf{n}_z$ refers to the vertical component of fluid velocity. However, this formulation is incomplete. Recall that the kinetic reaction rate has a surface area term ϕ_{opx}, which smoothly reduces the reaction to 0 as the opx mineral is exhausted. The equilibrium formulation no longer contains this factor, however, which leads to continual melt production despite

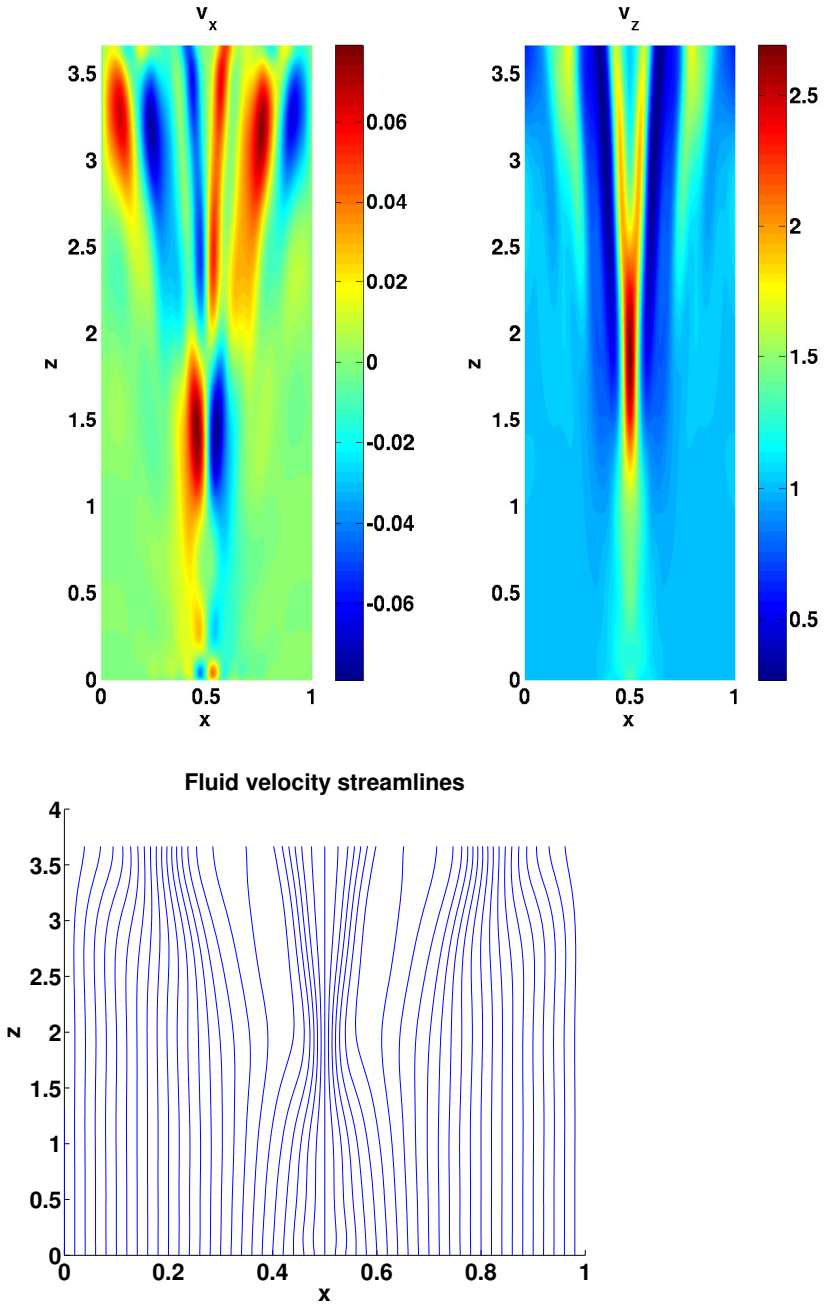

Figure 6.7: Top: fluid velocity vector $\mathbf{V}_f = (v_x, v_z)$, for profiles presented in Figure 6.6. For v_x, negative (positive) values represent leftwards (rightwards) flow. Before the channel bifurcates, the fluid tends to flow more inwards towards the channel, while after the bifurcation it flows outwards. The vertical component v_z resembles the porosity. Bottom: fluid velocity streamlines. The fluid tends to avoid low-porosity compaction regions.

there not being a dissolvable source (opx) remaining.

One fairly straightforward, if not naive, way to deal with this is to simply take

$$\Gamma_{opx}(x,z,t) = 0, \quad \text{if } \phi_{opx}(x,z,t) = 0, \tag{6.20}$$

and accepting Eq. (6.19) otherwise. As before we set any negative values of ϕ_{opx} to 0. In Figure 6.8, results are presented for the same problem as in Figure 6.6, but with $Da = \infty$ and Γ_{opx} modified as described in Eqs. (6.19)-(6.20). We observe that the two solution sets appear very similar to another, although not surprisingly, the equilibrium case contains deeper melt channels.

This is not to suggest, however, that such an approach is completely foolproof. By simply setting the reaction term to 0 on gridpoints where the opx is exhausted we have made an approximation to the true location of the channel interface, which currently prevents convergence of the kinetic to equilibrium versions of the code. In Figure 6.9 a comparison of varying Da values is explored, demonstrating that although the equilibrium solution closely resembles the kinetic, they are not in complete agreement. In future work we will explore more sophisticated models of tracking the $\phi_{opx} = 0$ regions in pursuit reconciling the two formulations.

6.6 Concluding Remarks

In this work, a high-order numerical scheme consisting of a discontinuous Galerkin finite element method and Fourier spectral method have been used to model melt generation via the reactive infiltration stability in an upwelling compacting mantle. The numerical discretization has been verified by matching growth rates predicted from a linear stability analysis to transient behavior of the scheme. Two non-dimensionalizations, one based upon a reactive time scale and one based upon an upwelling time scale, have been considered.

A strong correlation between mantle heterogeneity and channel formation has

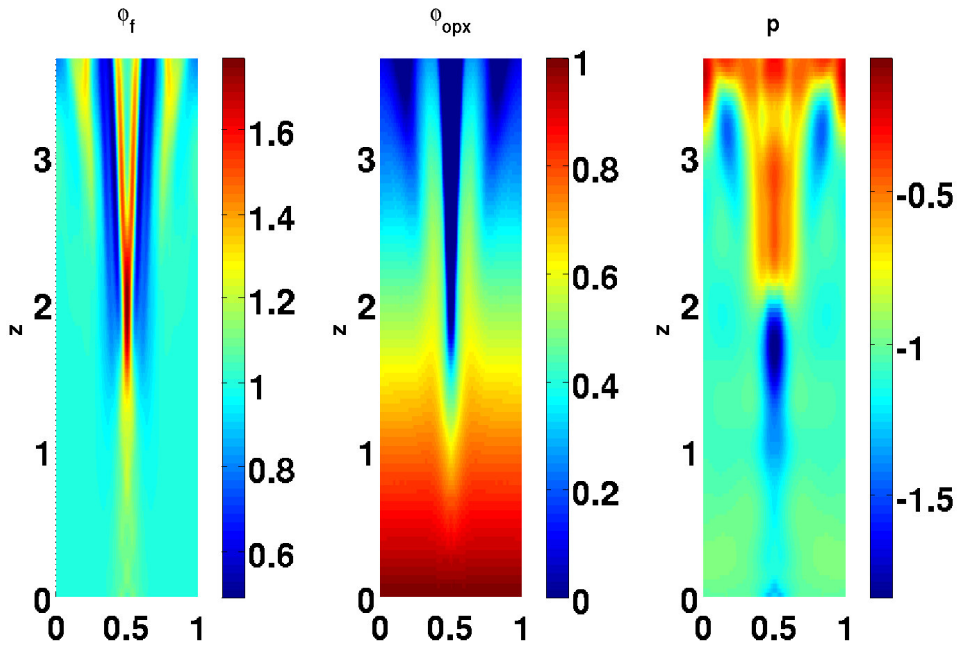

Figure 6.8: Transient solution with fixed boundary perturbation given in Eq. (6.18). Terminal time is $t = 2$. Reaction constant is $Da = \infty$. Numerical results are very similar to Figure 6.6, but are available at considerably less computational expense.

been discovered, aiding in the of magma transport in the Earth's mantle. Unlike previous models, we account for the dissolvable mineral opx present, and incorporate it as a surface area term in the reaction rate. From this we observe that channels bifurcate in the event of exhaustion of opx mineral, a previously unseen phenomenon.

In the limit of the reaction rate constant relative to melt velocity going to infinity (that is, the equilibrium formulation), we can reformulate the reaction term in a fashion that eliminates the opx-dependent surface area, effectively decoupling the system into the two variables ϕ_f and p. Still, the opx fraction plays a crucial role in determining where reaction must be turned off. Future areas of research should include a more precise treatment for the reaction rate in the equilibrium formulation, so as to ensure that reaction is not occurring when opx has been depleted.

Finally, the kinetic formulation is hindered by a restrictive time step that scales as the order of the reaction constant. Reconciling this issue, perhaps with a type of

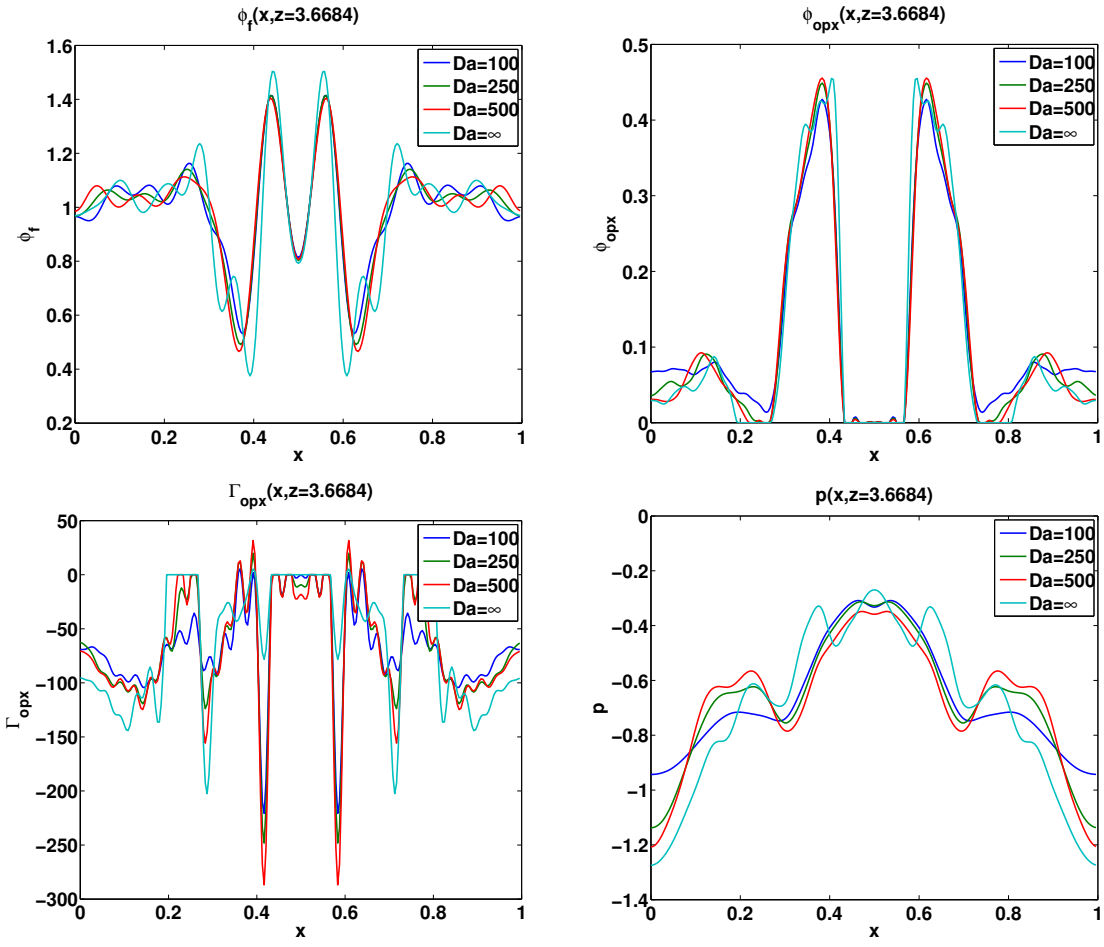

Figure 6.9: Comparison of varying Da behaviors at solution outflow $z \approx 3.66$, time $t = 2$, for transient solution with fixed boundary perturbation given in Eq. (6.18). The equilibrium solution closely resembles the kinetic.

semi-implicit time stepping scheme, is preferable for lowering computational expense. This improvement, coupled with a more careful treatment of opx-free regions in the equilibrium formulation mentioned above, will enable us to better demonstrate convergence of the two formulations as the reaction constant approaches infinity.

Bibliography

[1] E. Aharonov, M. Spiegelman, and P. Kelemen. Three-dimensional flow and reaction in porous media: Implications for the earth's mantle and sedimentary basins. *J. Geophys. Res.*, 102:821–834, 1997.

[2] E. Aharonov, J. Whitehead, P. B. Kelemen, and M. Spiegelman. Channeling instability of upwelling melt in the mantle. *J. Geophys. Res.*, 100:433–450, 1995.

[3] D. Arnold, F. Brezzi, B. Cockburn, and L. Marini. Unified analysis of discontinuous Galerkin methods for elliptic problems. *SIAM J. Numer. Anal.*, 39:1749–1779, 2002.

[4] J. Bear. *Dynamics of Fluids in Porous Media*. American Elsevier Pub. Co., 1972.

[5] D. Bercovici, Y. Ricard, and G. Schubert. A two-phase model for compaction and damage 1. General theory. *J. Geophys. Res.*, 106:8887–8906, 2001.

[6] J. Butcher. *The Numerical Analysis of Ordinary Differential Equations: Runge-Kutta and General Linear Methods*. John Wiley & Sons, 2003.

[7] M. H. Carpenter and C. A. Kennedy. Fourth-order 2n-storage Runge-Kutta schemes. Nasa tm 109112, NASA Langley Research Center, 1994.

[8] J. Chadam, D. Hoff, E. Merino, P. Ortoleva, and A. Sen. Reactive infiltration instability. *J. Appl. Math*, 36:207–221, 1986.

[9] B. Cockburn. Discontinuous Galerkin methods for convection-dominated problems. In *High-Order Methods for Computational Physics*. 1999.

[10] B. Cockburn, S. Hou, and C.-W. Shu. TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws iv: The multidimensional case. *Math. Comp.*, 54:545–581, 1990.

[11] B. Cockburn, S. Lin, and C.-W. Shu. TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws iii: One-dimensional systems. *J. Comput. Phys.*, 84:90–113, 1989.

[12] B. Cockburn and C.-W. Shu. The Runge-Kutta local projection p^1-discontinuous Galerkin method for scalar conservation laws. *RAIRO Modél. Math. Anal. Numér.*, 25:337–361, 1991.

[13] B. Cockburn and C.-W. Shu. TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws ii: General framework. *Math. Comp.*, 52:411–435, 1998.

[14] B. Cockburn and C.-W. Shu. TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws v: Multidimensional systems. *J. Comput. Phys.*, 141:199–224, 1998.

[15] M. J. Daines and D. L. Kohlstedt. The transition from porous to channelized flow due to melt/rock reaction during melt migration. *Geophys. Res. Lett.*, 21:145–148, 1994.

[16] P. Davis and P. Rabinowitz. *Methods of Numerical Integration*. Academic Press, 1975.

[17] M. Dumbser and M. Kaser. Arbitrary high order non-oscillatory finite volume schemes on unstructured meshes for linear hyperbolic problems. *J. Comput. Phys.*, 221:693–723, 2007.

[18] M. Dumbser, M. Kaser, V. A. Titarev, and E. F. Toro. Quadrature-free non-oscillatory finite volume schemes on unstructured meshes for nonlinear hyperbolic systems. *J. Comput. Phys.*, 226:204–243, 2007.

[19] D. Elthon and C. M. Scarfe. High-pressure phrase equilibria of a high-magnesia basalt and the genesis of primary oceanic basalts. *Am. Mineral.*, 69:1–15, 1984.

[20] L. C. Evans. *Partial Differential Equations*, pages 205–206. American Mathematical Society, 1998.

[21] G. Golub and J. Welsch. Calculation of gauss quadrature rules. *Math. Comp.*, 23:221–230, 1969.

[22] D. Gottlieb and J. Hesthaven. Spectral methods for hyperbolic problems. *J. Comp. Appl. Math.*, 128:83–131, 2001.

[23] E. Hairer, S. N. rsett, and G. Wanner. *Solving Ordinary Differential Equations I. Nonstiff Problems.* Springer Series in Comput. Mathematics. Springer-Verlag, 1987.

[24] E. Hairer and G. Wanner. *Solving Ordinary Differential Equations II. Stiff and Differential-Algebraic Problems.* Springer Series in Comput. Mathematics. Springer-Verlag, 1991.

[25] A. Harten, B. Engquist, S. Osher, and S. R. Chakravarthy. Uniformly high order essentially non-oscillatory schemes, iii. *J. Comput. Phys.*, 71:231–303, 1987.

[26] J. Hesthaven. From electrostatics to almost optimal nodal sets for polynomial interpolation in a simplex. *SIAM J. Numer. Anal.*, 35:655–676, 1998.

[27] J. Hesthaven, S. Gottlieb, and D. Gottlieb. *Spectral Methods for Time-Dependent Problems.* Cambridge University Press, 2006.

[28] J. Hesthaven and T. Warburton. *Nodal Discontinuous Galerkin Methods: Algorithms, Analysis, and Applications*. Springer, 2008.

[29] G. Jiang and C.-W. Shu. On a cell entropy inequality for discontinuous Galerkin methods. *Math. Comp.*, 62:531–538, 1994.

[30] G.-S. Jiang and C.-W. Shu. Efficient implementation of weighted eno schemes. *J. Comput. Phys.*, 126:202–228, 1996.

[31] P. B. Kelemen. The origin of the land under the sea. *Sci. Am.*, pages 52–57, Feb. 2009.

[32] P. B. Kelemen, G. Hirth, N. Shimizu, M. Spiegelman, and H. Dick. A review of melt migration processes in the adiabatically upwelling mantle beneath oceanic spreading ridges. *Phil. Trans. R. Soc. Lond.*, 355:282–318, 1997.

[33] P. B. Kelemen, J. A. Whitehead, E. Aharonov, and K. A. Jordahl. Experiments on flow focusing in soluble porous media, with applications to melt extraction from the mantle. *J. Geophys. Res.*, 100:475–496, 1995.

[34] R. J. Kinzler and T. L. Grove. Primary magmas of mid-ocean ridge basalts. 2 applications. *J. Geophys. Res.*, 97:6907–6926, 1992.

[35] E. Klein and C. H. Langmuir. Global corrections of ocean ridge basalt chemistry with axial depth and crustal thickness. *J. Geophys. Res.*, 92:8089–8115, 1987.

[36] R. LeVeque. *Finite Volume Methods for Hyperbolic Problems*. Cambridge University Press, 2002.

[37] X.-D. Liu, S. osher, and T. Chan. Weighted essentially non-oscillatory schemes. *J. Comput. Phys.*, 115:200–212, 1994.

[38] H. Luo, J. D. Baum, and R. Lohner. A Hermite WENO-based limiter for discontinuous Galerkin method on unstructured grids. *J. Comput. Phys.*, 225:686–713, 2007.

[39] D. McKenzie. The generation and compaction of partially molten rock. *J. Petrol.*, 25:713–765, 1984.

[40] M. J. O'Hara. Primary magmas and the origin of basalts. *Scot. J. Geol.*, 1:19–40, 1965.

[41] C. F. Ollivier-Gooch. Quasi-eno schemes for unstructured meshes based on unlimited data-dependent least-squares reconstruction. *J. Comput. Phys.*, 133:6–17, 1997.

[42] P. Ortoleva, E. Merino, C. Moore, and J. Chadam. Geochemical self-organization, i, reaction-transport feedbacks. *Am. J. sci.*, 287:979–1007, 1987.

[43] J. Qiu and C.-W. Shu. Hermite WENO schemes and their application as limiters for Runge-Kutta discontinuous Galerkin method: One dimensional case. *J. Comput. Phys.*, 212:540–565, 2003.

[44] J. Qiu and C.-W. Shu. A comparison of troubled cell indicators for Runge-Kutta discontinuous Galerkin methods using WENO limiters. *SIAM J. Sci. Comput.*, 27:995–1013, 2005.

[45] J. Qiu and C.-W. Shu. Hermite WENO schemes and their application as limiters for Runge-Kutta discontinuous Galerkin method ii: Two dimensional case. *Comput. & Fluids*, 34:642–663, 2005.

[46] J. Qiu and C.-W. Shu. Runge-Kutta discontinuous Galerkin method using WENO limiters. *SIAM J. Sci. Comput.*, 26:907–929, 2005.

[47] W. Reed and T. Hill. Triangular mesh methods for the neutron transport equation. Technical Report LA-UR-73-479, Los Alamos Scientific Laboratory, 1973.

[48] D. R. Scott and D. J. Stevenson. Magma ascent by porous flow. *J. Geophys. Res.*, 91:9283–9296, 1986.

[49] C.-W. Shu. TVB uniformly high-order schemes for conservation laws. *Math. Comp.*, 49:105–121, 1987.

[50] C.-W. Shu. TVD time-discretizations. *SIAM J. Sci. Statist. Comput.*, 9:1073–1084, 1988.

[51] C.-W. Shu. Essentially non-oscillatory and weighted essentially non-oscillatory schemes for hyperbolic conservation laws. Technical Report ICASE Report No. 97-65, NASA Langley Research Center, 1997.

[52] C.-W. Shu and S. Osher. Efficient implementation of essentially non-oscillatory shock-capturing schemes. *J. Comput. Phys.*, 77:439–471, 1988.

[53] N. H. Sleep. Segregation of magma from a mostly crystalline mush. *Geol. Soc. Am. Bull*, 85:1225–1232, 1974.

[54] M. Spiegelman. Flow in deformable porous media, 1, simple analysis. *J. Fluid. Mech.*, 247:39–63, 1993.

[55] M. Spiegelman and P. B. Kelemen. Extreme chemical variability as a consequence of channelized melt transport. *Geochem. Geophys. Geosyst.*, 4(7), 2001.

[56] M. Spiegelman, P. B. Kelemen, and E. Aharonov. Causes and consequences of flow organization during melt transport: The reaction infiltration instability in compactible media. *J. Geo. Res.*, 106:2061–2077, 2001.

[57] E. Stolper. A phase diagram for mid-ocean ridge basalts: Preliminary results and implications for petrogenesis. *Contrib. Mineral. Petrol.*, 74:13–27, 1980.

[58] G. Szego. *Orthogonal Polynomials*. American Mathematical Society, 1981.

[59] L. Trefethen. *Spectral Methods in Matlab*. SIAM Publishing, 2000.