

# A multi-numeric method for parabolic problems using an adaptive region-swapping approach

Joseph Huchette (mentored by Dr. Beatrice Riviere)  
Rice University

May 14, 2012

## Abstract

Attempts to approximately solve parabolic convection-diffusion partial differential equations accurately with a minimum of computational cost motivates the investigation of a coupled multi-numeric method that takes advantage of an adaptive domain-partitioning approach. In this work, the finite volume method—a low cost, low accuracy method—is coupled with the discontinuous Galerkin method—a high cost, high accuracy method. For a fixed grid, the subsets of the domain on which each method is applied change at each time-step, with the intention of applying the more accurate method where necessary and the less costly method wherever else. Implementing this method for convection-dominated problems yields results that are qualitatively similar to that yielded by the sole application of the more accurate solution and that preserve the expected numerical convergence rates.

## 1 Introduction

Applications in the oil and gas industry often necessitate an accurate modeling of convection-dominated parabolic problems. The solutions for these problems often resemble a wave or crest traveling through the domain. The finite volume (FV) method is frequently applied to these problems because of its low computational cost, flux conservation between neighboring elements, and adaptability to arbitrary domains. First order FV methods are widely used in reservoir simulators in the oil industry, yielding what is often an undesirably low error bound. Alternatively, the discontinuous Galerkin (DG) method can be made to have an arbitrarily high-order error bound, offering a high degree of accuracy that is necessarily tempered by a proportionally higher computational cost.

The goal of this paper is to develop a method for approaching these problems using a coupled multi-numeric approach, mixing FV and DG methods so that the advantages of each are exploited and the disadvantages minimized. More succinctly, the goal is to solve the problem with the highest accuracy at the lowest cost. The most obvious way to approach this is to apply each method to a discrete partition of the domain and then couple them together to produce a consistent solution. Due to the time-dependent nature of this problem, however, in general this optimal partition of the domain will not remain static through time, but instead will change with each time-step.

The finite volume and discontinuous Galerkin methods have been extensively studied and there are numerous texts available; [2] and [3] were used as the primary DG and FV references, respectively. There are also literature investigating the coupling of the two methods for elliptic problems; for example, [5] and [6]. Previous literature [1] has established a theoretical foundation for a coupled DG/FV scheme for a one-dimensional elliptic problem.

The rest of this paper will consider the one-dimensional formulation of the problem. It will develop a coupled DG/FV method for solving a one-dimensional, parabolic convection-diffusion boundary value

problem with Dirichlet boundary conditions of the form

$$\begin{aligned} \frac{\partial u(x, t)}{\partial t} - \kappa \frac{\partial^2 u(x, t)}{\partial x^2} + \varphi \frac{\partial u(x, t)}{\partial x} &= f(x, t), \quad x \in [a, b], \quad t \in [t_0, t_F] \\ u(a, t) &= \alpha(t), \quad u(b, t) = \beta(t), \quad u(x, t_0) = u^0(x) \end{aligned} \quad (1)$$

allowing the domain to be arbitrarily partitioned into discrete and non-intersecting DG and FV sections. The solution vector is  $u$ ; the driving function is  $f \in C([t_0, t_F] \times (a, b))$ ;  $\kappa \in \mathbb{R}^+$  and  $\varphi \in \mathbb{R}^+$  are the coefficients of the diffusion and convection terms, respectively;  $\alpha \in \mathbb{R}$  and  $\beta \in \mathbb{R}$  are the boundary condition values at endpoints  $a$  and  $b$ ; and the initial condition is  $u^0$ , which lives in the appropriate solution space to be defined later. The implementation introduced will choose a DG method of degree  $r = 2$ , although the framework for a higher-degree method will be covered. Also, the implementation will develop a similar solver that changes the locations of the interfaces between the DG and FV sections at each time-step, effectively changing the partitions between the two methods. Henceforth, this adaptive domain-partitioning approach will be called “region-swapping.”

## 2 Notation and Scheme

A region is the smallest discrete element (interval) of the domain over which the PDE is solved, and is named  $\gamma_i$  with a global numbering system. A section is a set of regions such that all regions within it are solved over with the same method and such that the set is contiguous. A section is denoted by  $FV_k$  or  $DG_k$  if it is the  $k$ th FV or DG section, respectively. A sectioning is the set of all sections in the domain. Define  $M^{DG}$  as the number of DG sections, and  $M^{FV}$  as the number of FV sections, and notate  $N^{FV_k}$  and  $N^{DG_k}$  as the number of regions in  $FV_k$  and  $DG_k$ , respectively. The number  $N$  without any superscript denotes the total number of regions in the domain. Throughout  $u$  will be used to denote the exact solution, while  $u_h$  denotes the approximate solution for a given partition.

### 2.0.1 FV Method Notation

Consider the FV section  $FV_k = \{\gamma_1^{FV_k}, \dots, \gamma_{N^{FV_k}}^{FV_k}\}$  with  $N^{FV_k}$  regions. The endpoints of  $\gamma_i$  are denoted by  $x_{j-\frac{1}{2}}^{FV_k}$  and  $x_{j+\frac{1}{2}}^{FV_k}$ . Similarly, as necessitated by the FV method, take an additional point within the region  $\gamma_{n+j}^{FV_k}$  denoted by  $x_{j+1}^{FV_k}$  for  $j \in \{0, \dots, N^{FV_k} - 1\}$ . The FV method is a first order method, so it yields a constant solution for each region  $\gamma_j^{FV_k}$ ; this will be denoted by  $u_j$ . The width of the region  $\gamma_j^{FV_k}$  is  $h_j^{FV_k} = x_{j+\frac{1}{2}}^{FV_k} - x_{j-\frac{1}{2}}^{FV_k}$ . In general these widths can be of different size for each FV region, although for the simplicity of the numerical results a uniform mesh will be used, and so we will take  $h_i^{FV_k} = h$  later in the text when numerics are introduced. On occasion the alternative notation  $v_j = v(x), \forall x \in \gamma_j^{FV_k}$ , will be used, which is valid since  $v$  is piecewise constant over the FV regions.

### 2.0.2 DG Method Notation

Consider the DG section  $DG_k = \{\gamma_1^{DG_k}, \dots, \gamma_{1+N^{DG_k}}^{DG_k}\}$  with  $N^{DG_k}$  regions. Take the endpoints of  $\gamma_j^{DG_k}$  as  $x_{j-1}^{DG_k}$  and  $x_j^{DG_k}$ . The DG method of order  $r$  yields a polynomial of degree  $r$  for each region; denote the coefficients of this polynomial with  $u_j$  and reference the  $q$ th element of this local solution with  $u_{j,q}$ . For this implementation, it is assumed that the widths  $h_j^{DG_k} = x_j^{DG_k} - x_{j-1}^{DG_k}$  are uniform, so all  $h_j^{DG_k} = h$ .

Notate the values achieved at  $x_j^{DG_k}$  from the local solution over region  $\gamma_j^{DG_k}$  as  $u(\bar{x}_j^{DG_k})$  and from the local solution over region  $\gamma_{j+1}^{DG_k}$  as  $u(\bar{x}_j^{DG_k})$ . The so-called jump value at node  $j$  is defined as  $\left[ u(\bar{x}_j^{DG_k}) \right] = u(\bar{x}_j^{DG_k}) - u(\bar{x}_j^{DG_k})$ . Similarly, the average value at node  $j$  is defined as  $\left\{ u(\bar{x}_j^{DG_k}) \right\} = \frac{1}{2} \left( u(\bar{x}_j^{DG_k}) + u(\bar{x}_j^{DG_k}) \right)$ .

The bases functions for the DG method used in this context are piecewise discontinuous polynomial. For the quadratic degree, there are three bases functions for each region  $\gamma_i^{DG_k}$ . They are polynomials over the region  $\gamma_i^{DG_k}$  and take value zero outside the region.

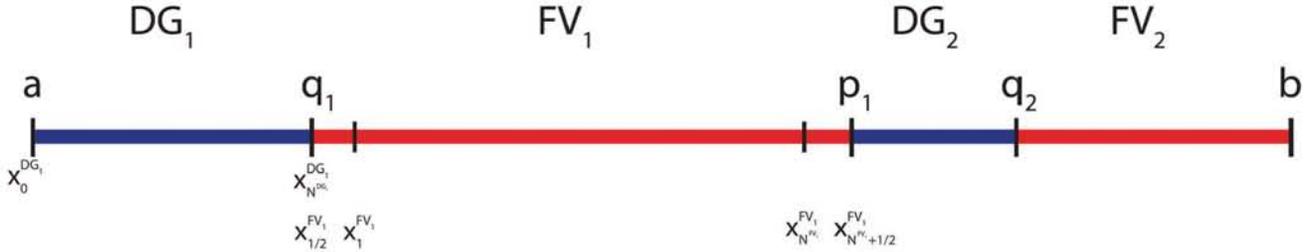


Figure 1: Illustration of a example domain. The domain is split into two DG and two FV regions with interfaces  $q_1$ ,  $q_2$ , and  $p_1$ . The nodes for  $DG_1$  and  $FV_1$  closest to the interfaces are labelled to show the notation used throughout the remainder of the text.

### 2.0.3 Interface Notation

Define  $P$  as the set of interfaces between sections where a  $FV$  section is to the left of a  $DG$  section and  $Q$  as the set of those interfaces with  $DG$  on the left and  $FV$  on the right (see Figure 1). Take that  $w_{DG}(x)$  denotes  $w$  evaluated on the adjacent DG section to  $x$  and that  $w_{FV}(x)$  denotes  $w$  evaluated on the adjacent FV section to  $x$ . Take  $\mathbb{P}_r(\gamma_i)$  to denote the space of piecewise discontinuous polynomials of degree  $r$  over region  $\gamma_i$ .

### 2.1 Scheme

The scheme applied to problem (1) takes the form:

Find  $u_h \in V = \left\{ v : v|_{\gamma_i^{DG_k}} \in \mathbb{P}_r(\gamma_i^{DG_k}) \forall DG \text{ regions } \gamma_i^{DG_k}, v|_{\gamma_j^{FV_k}} \in \mathbb{P}_0(\gamma_j^{FV_k}) \forall FV \text{ regions } \gamma_j^{FV_k} \right\}$   
such that

$$\int_a^b \frac{\partial u_h}{\partial t} v \, dx + \sum_{k=1}^{M^{DG}} (a_{\kappa}^{DG_k}(u_h, v) + a_{\varphi}^{DG_k}(u_h, v)) + \sum_{k=1}^{M^{FV}} (a_{\kappa}^{FV_k}(u_h, v) + a_{\varphi}^{FV_k}(u_h, v))$$

$$+ \sum_{\xi \in P \cup Q} (d_{\kappa}^{\xi}(u_h, v) + d_{\varphi}^{\xi}(u_h, v)) + \mathcal{L}(u_h, v) + \mathcal{R}(u_h, v) \tag{2}$$

$$= \sum_{k=1}^{M^{DG}} b^{DG_k}(v) + \sum_{k=1}^{M^{FV}} b^{FV_k}(v) + L_{\kappa}(v) + L_{\varphi}(v) + R_{\kappa}(v)$$

for all  $v \in V$ .

We now define the bilinear forms introduced in (2).

### 2.1.1 Diffusion Term and Right-Hand Side Term

First, we define the DG forms.

Take a DG section  $DG_k$  with nodes  $x_0^{DG_k}, \dots, x_{N^{DG_k}}^{DG_k}$ . The  $b^{DG_k}$  form is derived by integrating the right-hand side of (1) over the entire section:

$$b^{DG_k}(v) = \int_{x_0^{DG_k}}^{x_{N^{DG_k}}^{DG_k}} f v \, dx$$

Integrating the diffusion term over each region  $\gamma_k^{DG_k}$  and summing all regions in  $DG_k$  in the usual fashion yields the basic bilinear form for the diffusion term:

$$\begin{aligned} a_\kappa^{DG_k}(u, v) &= \kappa \sum_{i=1}^{N^{DG_k}} \int_{x_{i-1}^{DG_k}}^{x_i^{DG_k}} u' v' \, dx - \kappa \sum_{i=1}^{N^{DG_k}-1} \left\{ u' \left( x_i^{DG_k} \right) \right\} \left[ v \left( x_i^{DG_k} \right) \right] \\ &\quad - \epsilon \kappa \sum_{i=1}^{N^{DG_k}-1} \left\{ v' \left( x_i^{DG_k} \right) \right\} \left[ u \left( x_i^{DG_k} \right) \right] + \frac{\sigma}{h} \sum_{i=1}^{N^{DG_k}-1} \left[ u \left( x_i^{DG_k} \right) \right] \left[ v \left( x_i^{DG_k} \right) \right] \end{aligned}$$

In this scheme,  $\sigma$  is the penalty term introduced in the DG method; it is introduced to “penalize” discontinuity in the approximate solution to best match the actual solution (which is continuous). The parameter  $\epsilon \in \{-1, 1\}$  signifies that either the symmetric interior penalty Galerkin (SIPG) method is used ( $\epsilon = -1$ ) or that the nonsymmetric interior penalty Galerkin (NIPG) method is used ( $\epsilon = 1$ ) [2]. For the NIPG method, the penalty parameter is taken to be  $\sigma = 1$ ; for the SIPG method, it should be taken sufficiently large.

If one endpoint of the section lies on the boundary of the domain, then the remaining terms form the boundary forms.

If  $x_0^{DG_k} = a$  (i.e. the first DG region is on the left boundary of the domain), then

$$\begin{aligned} \mathcal{L}(u, v) &= \kappa u'(a)v(a) + \epsilon \kappa v'(a)u(a) + \frac{\sigma}{h} u(a)v(a) \\ L_\kappa(v) &= \kappa \alpha \epsilon v'(a) + \frac{\sigma}{h} \alpha v(a) \end{aligned}$$

Similarly, if  $x_{N^{DG_k}}^{DG_k} = b$  (i.e. the last DG region is on the right boundary of the domain), then

$$\begin{aligned} \mathcal{R}(u, v) &= -\kappa u'(b)v(b) - \epsilon \kappa v'(b)u(b) + \frac{\sigma}{h} u(b)v(b) \\ R_\kappa(v) &= -\beta \epsilon \kappa v'(b) + \frac{\sigma}{h} \beta v(b) \end{aligned}$$

All other terms left are included in the interface terms, which will be defined more meaningfully at the end of the section, incorporating similar terms from FV sections.

Next, we define the FV forms for a given section  $FV_k$ :

$$\begin{aligned} a_\kappa^{FV_k}(u, v) &= \kappa \sum_{i=2}^{N^{FV_k}-1} \left( \frac{-u \left( x_{i+1}^{FV_k} \right) + u \left( x_i^{FV_k} \right)}{h_{i+\frac{1}{2}}} v_i + \frac{u \left( x_i^{FV_k} \right) - u \left( x_{i-1}^{FV_k} \right)}{h_{i-\frac{1}{2}}} v_i \right) \\ &\quad - \kappa \frac{-u \left( x_2^{FV_k} \right) + u \left( x_1^{FV_k} \right)}{h_{\frac{3}{2}}} v_1 + \kappa \frac{u \left( x_{N^{FV_k}}^{DG_k} \right) - u \left( x_{N^{FV_k}-1}^{FV_k} \right)}{h_{N^{FV_k}-\frac{1}{2}}} v_{N^{FV_k}} \\ b^{FV_k}(v) &= \int_{x_{\frac{1}{2}}^{FV_k}}^{x_{N^{FV_k}+\frac{1}{2}}^{FV_k}} f v \, dx \end{aligned}$$

If either section boundary lies on the boundary of the domain, the boundary forms are appropriately defined.

If  $x_{\frac{1}{2}}^{FV_k} = a$ , then

$$\begin{aligned}\mathcal{L}(u, v) &= \kappa \frac{u(x_1^{FV_k})}{h_1} v_1 \\ L_\kappa(u, v) &= \kappa \frac{\alpha}{h_1} v_1\end{aligned}$$

Similarly, if  $x_{N^{FV_k} + \frac{1}{2}}^{FV_k} = b$ , then

$$\begin{aligned}\mathcal{R}(u, v) &= \kappa \frac{u(x_{N^{FV_k}}^n)}{h_{N^{FV_k}}} v_{N^{FV_k}} \\ R_\kappa(v) &= \kappa \frac{\beta}{h_M} v_{N^{FV_k}}\end{aligned}$$

It is now appropriate to define the interface forms since all terms have been introduced. They take the form

$$d_\kappa^\xi(u, v) = \frac{\kappa}{\tilde{h}_\xi} (u_{DG}(\xi) - u_{FV}(\xi)) (v_{DG}(\xi) - v_{FV}(\xi))$$

where

$$\tilde{h}_\xi = \begin{cases} x_{N^{FV_k}}^{FV_k} - \xi & \text{if } \xi \in P \\ -x_1^{FV_k} + \xi & \text{if } \xi \in Q \end{cases}$$

and  $FV_k$  is the FV section adjacent to  $\xi$ .

### 2.1.2 Convection Term

We first consider an arbitrary DG region  $DG_k$  with  $N^{DG_k}$  regions and nodes  $x_0^{DG_k}, \dots, x_{N^{DG_k}}^{DG_k}$ . Integrating the new convection term over the interval containing region  $\gamma_j^{DG_k}$  leads to

$$\int_{x_{j-1}^{DG_k}}^{x_j^{DG_k}} \varphi \frac{\partial u(x)}{\partial x} v dx = \varphi \left( u \left( \bar{x}_j^{DG_k} \right) v \left( \bar{x}_j^{DG_k} \right) - u \left( \hat{x}_{j-1}^{DG_k} \right) v \left( \hat{x}_{j-1}^{DG_k} \right) \right) - \varphi \int_{x_{j-1}}^{x_j} uw' dx$$

Summing the first boundary term over all intervals from 1 to  $N^{DG_k}$  leads to

$$\begin{aligned}& \sum_{j=1}^{N^{DG_k}} \varphi \left( u \left( \bar{x}_j^{DG_k} \right) v \left( \bar{x}_j^{DG_k} \right) - u \left( \hat{x}_{j-1}^{DG_k} \right) v \left( \hat{x}_{j-1}^{DG_k} \right) \right) \\ &= -\varphi \sum_{j=1}^{N^{DG_k}} u \left( \hat{x}_j^{DG_k} \right) \left[ v \left( x_j^{DG_k} \right) \right] + \varphi u \left( \bar{x}_{N^{DG_k}}^{DG_k} \right) v \left( \bar{x}_{N^{DG_k}}^{DG_k} \right) - \varphi u \left( \hat{x}_0^{DG_k} \right) v \left( \hat{x}_0^{DG_k} \right)\end{aligned}$$

where  $u \left( \hat{x}_j^{DG_k} \right)$  denotes the upwind approximation for  $u \left( x_j^{DG_k} \right)$ , chosen for stability reasons to be  $u \left( x_j^{DG_k} \right)$  [7]. This yields the bilinear form:

$$a_\varphi^{DG_k}(u, v) = \sum_{j=1}^{N^{DG_k}} \int_{x_{j-1}^{DG_k}}^{x_j^{DG_k}} uw' dx - \varphi \sum_{j=1}^{N^{DG_k}} u \left( \bar{x}_j^{DG_k} \right) \left[ v \left( x_j^{DG_k} \right) \right]$$

If  $x_0^{DG_k} = a$ , then

$$L_\varphi(v) = \varphi\alpha v \left( x_0^{DG_k} \right);$$

similarly, if  $x_{N^{DG_k}}^{DG_k} = b$ , then

$$R_\varphi(v) = -\varphi\beta v \left( x_{N^{DG_k}}^{DG_k} \right).$$

Next, we consider an arbitrary FV section  $FV_k$  with  $N^{FV_k}$  regions. Integrating the convection term over each region from  $x_{j-\frac{1}{2}}^{FV_k}$  to  $x_{j+\frac{1}{2}}^{FV_k}$  leads to

$$\int_{x_{j-\frac{1}{2}}^{FV_k}}^{x_{j+\frac{1}{2}}^{FV_k}} \varphi u' v_j = \varphi \left( u \left( x_{j+\frac{1}{2}}^{FV_k} \right) - u \left( x_{j-\frac{1}{2}}^{FV_k} \right) \right) v_j$$

Using an upwind approximation for stability, take the value of  $u$  at the endpoints to be the value of  $u$  at the node directly to the left. Summing over each region yields a bilinear form:

$$a_\varphi^{FV_k}(u, v) = \sum_{j=2}^{N^{FV_k}} \varphi \left( u \left( x_j^{FV_k} \right) - u \left( x_{j-1}^{FV_k} \right) \right) v_j.$$

If  $x_{\frac{1}{2}}^{FV_k} = a$ , then

$$L_\varphi(v) = \varphi\alpha v \left( x_{\frac{1}{2}}^{FV_k} \right)$$

Since an upwind approximation is used, there is no contribution to the right-hand side of (2) for the convection term at the endpoint  $b$ .

It is now appropriate to define the interface forms as:

$$d_\varphi^\xi(u, v) = \begin{cases} \varphi u_{DG}(\xi) v_{DG}(\xi) - \varphi u_{FV}(\xi) v_{DG}(\xi) & \text{if } \xi \in Q \\ -\varphi u_{DG}(\xi) v_{DG}(\xi) & \text{if } \xi \in P \end{cases}$$

## 2.2 Time Discretization

For this paper, a simple backward Euler time-stepper was chosen, although any similarly appropriate method would also work. The approach is to semi-discretize the problem (1) in space into a linear system that can be solved using established methods. The stiffness matrix  $\mathbf{A}$  and load vector  $\mathbf{b}$  are constructed to discretize the problem as laid out in the scheme above. Denote the set of unknowns as the vector  $\mathbf{u}_h$ . Under this spatial discretization, the backward Euler time-stepper yields the iterative step

$$\mathbf{u}_h^n = \left( \frac{\mathbf{M}}{\Delta t} + \mathbf{A} \right)^{-1} \left( \mathbf{M} \frac{\mathbf{u}_h^{n-1}}{\Delta t} + \mathbf{b}^n \right) \quad (3)$$

where  $\mathbf{M}$  is the mass matrix arising in the usual form. The initial condition does not take interface contributions, being just a  $L^2$  projection of  $u^0$  onto  $V$ , and as such can be constructed in the usual way by concatenating local portions together; a similar approach holds for construction of the mass matrix.

## 3 Adaptive Region-Swapping Approach

The main goal of this research is to develop a consistent and accurate solver for a class of parabolic partial differential equations that couples the discontinuous Galerkin and finite volume method and that is capable of accommodating an arbitrary partition between the two methods at each time-step. Above a reliable solver

for the problem with an arbitrary partition between the two methods was established; the next step is to develop a reliable way for switching this partition at each time-step without undue loss of accuracy.

The approach will deal with a simpler form of the general problem, where each region is taken to be of the same size  $h$ . Relatedly, the interior points of the FV regions are chosen to be the midpoints of their respective region, for added uniformity for the region-swapping implementation. Additionally, the simplified problem will deal with cases where domain can be split arbitrarily between the two methods in any way such that there are a maximum of two FV sections and a single DG section. This is motivated by the fact that the DG region will follow the solution front for a convection-dominated problem. Consider the domain  $[a, b]$  discretized into  $N$  uniformly-sized regions  $\gamma_1, \dots, \gamma_N$ . It is assumed that the sections are of the form  $FV_1 = \{\gamma_j : j \in \mathbb{Z}, 1 \leq j \leq I_1 - 1\}$ ,  $DG = \{\gamma_j : j \in \mathbb{Z}, I_1 \leq j \leq I_2\}$ ,  $FV_2 = \{\gamma_j : j \in \mathbb{Z}, I_2 + 1 \leq j \leq N\}$ , where  $I_1, I_2 \in \{1, \dots, N\}$ .

### 3.1 Criteria for Region-Swapping

For this implementation, a certain user-defined tolerance term  $\mu$  is taken by the solver that determines the partition to be used at a particular time-step  $n + 1$ , based on this value  $\mu$  and the previous solution vector  $\mathbf{u}^n$  and accompanying sectioning. The value  $\mu$  specifies the minimum “slope” value used to choose the DG section boundaries  $I_1$  and  $I_2$ , where the term “slope” is used loosely to mean the difference quotient of the approximate solution  $u^n$  across a region  $\gamma_i$ . This value is referred to as the difference quotient, denoted by  $\chi(i)$  for region  $\gamma_i$ , and is straight forward to calculate, although there has to be some methodical way of choosing the values to calculate the quotient. Throughout this section, we denote the endpoints of  $\gamma_i$  by  $y_{i-1}$  and  $y_i$ .

#### 3.1.1 Calculating the Difference Quotient

When calculating the difference quotient, the aim is to most accurately gauge the change in the value of the approximate solution  $u_h^n$  over the region, using the information available in the solution vector and the boundary values. There are multiple ways to calculate this; one possible solution (used in the implementation) is shown pictorially in Figure 2. Referencing the figure,  $\chi(j) = \left. \frac{d\Psi_j}{dx} \right|_{\gamma_j}$ .

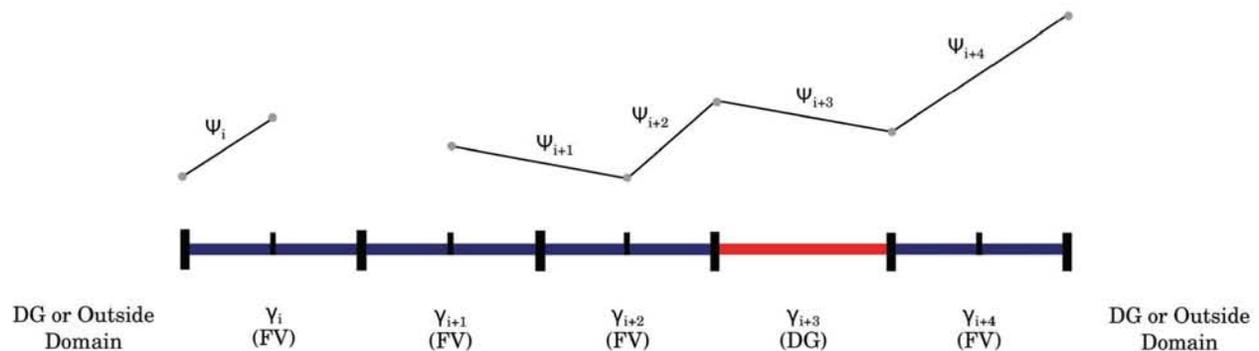


Figure 2: Depiction of example discretization with each possible region configuration for the calculation of the difference quotient. Also included are the linear interpolants between the two values used for each region; the difference quotient is simply the slope of this interpolant.

The implementation assigns the DG section such that it included all possible regions that could benefit from the increased accuracy of the DG method; as such, the indices  $i_1$  and  $i_2$  are chosen by

$$I_1 = \min\{i : i \in \mathbb{Z}, 1 \leq i \leq N, \chi(i) \geq \mu\}$$

$$I_2 = \max\{i : i \in \mathbb{Z}, 1 \leq i \leq N, \chi(i) \geq \mu\}$$

Consider a sectioning  $FV_1^n, DG^n, FV_2^n$ , the corresponding solution  $u$  at time-step  $n$ , and a different sectioning at time-step  $n + 1$  with similar sets  $FV_1^{n+1}, DG^{n+1}, FV_2^{n+1}$  and approximate solution  $\hat{u}$ . When dealing with these two sectionings, there are three possible cases: a single region  $\gamma_i$  can stay associated with the same method, it can switch from the DG section to a FV section, or it can switch from a FV section to the DG section. The first case is trivial, and  $\hat{u}_i = u_i$ . The other two cases are more subtle, and, when dealing with a DG method of degree  $r$ , require some kind of mapping from  $\mathbb{R}^{r+1} \rightarrow \mathbb{R}$  and  $\mathbb{R} \rightarrow \mathbb{R}^{r+1}$ , respectively, that will be developed shortly.

### 3.1.2 A note on the selection of $\mu$

It is important to note that the overarching goals are both to cut computational costs as well as maintain as much accuracy as possible. As such, it is often not optimal to choose an unduly large value for  $\mu$  in an attempt to minimize the number of DG regions. The new sections are chosen at a particular time-step based on the nature of the solution at the previous time-step. The problems treated are wave-like in nature, meaning that they traverse or travel through the domain as a single unit. As a result, unless some a priori knowledge of the “speed” of the wave is known or can be accurately inferred from the solutions at previous time-steps (which is not assumed for this implementation), it is only possible to base the new sections on information at the old time-step, and choose the  $\mu$  value conservatively enough so that the new DG section  $DG^{n+1}$  includes all  $\gamma_i$  that would benefit from the added accuracy of the DG method. For sufficiently “slow” or well behaved solutions (an admittedly vague and ill-defined condition), a value of  $\mu = 0.5$  was sufficient for most examples considered, but it was of course possible to craft examples that necessitated a smaller value. This “traveling” nature of the solutions treated also means that treating smaller DG sections does not necessarily translate to a more accurate solution; a sufficient “cushion” must be incorporated via the designated  $\mu$  value to ensure that the new DG sections include all regions that reasonably should be included at the new time-step. As mentioned, some kind of “wave speed” information would be supremely useful and could make the method even more cost-effective, but this avenue was not pursued.

## 3.2 DG $\rightarrow$ FV Region-Swapping

The first case considered is that which sees a DG region at one time-step become a FV region at the next; namely, when  $\gamma_i \in DG^n$  and  $\gamma_i \in FV_1^{n+1} \cup FV_2^{n+1}$ , where the superscripts denote the sectioning at the corresponding time-step.

In general, for a DG method of degree  $p$ , the solution on each region  $\gamma_i$  is approximated by a polynomial of degree  $r$ . Since the FV solution is piecewise constant, take this value to be the average value of the approximate DG solution over the entire region; namely,

$$u_h^{n+1}|_{\gamma_i} = \frac{1}{y_i - y_{i-1}} \int_{y_{i-1}}^{y_i} u_h^n|_{\gamma_i} dx$$

## 3.3 FV $\rightarrow$ DG Region-Swapping

The case when  $\gamma_i$  is a FV region at time-step  $n$  and becomes a DG region at time-step  $n + 1$  (i.e.  $\gamma_i \notin DG^n$  and  $\gamma_i \in DG^{n+1}$ ) is much more subtle than the other two possibilities; a way to best translate the single piece of information in  $u_i^n$  into the  $r + 1$  pieces of information in  $\hat{u}_i^n$  must be chosen. After mapping the region  $\gamma_i$  to the reference interval  $[-1, 1]$ , the goal is to construct a piecewise discontinuous polynomial that is zero outside the region and takes the form  $\sum_{j=0}^r a_{i_j} \bar{x}^j$ ,  $\bar{x} \in [-1, 1]$ . The simplest way to do this would be to simply set  $a_{i_0} = u_i^k$ ,  $a_{i_j} = 0$  for  $j = 2, \dots, r$ . In other words, the approximating polynomial over  $\gamma_i$  is simply the constant value of  $u_i^n$ . Although this is perfectly valid, it would be preferable to preserve

as much of the added accuracy gained by using the DG method in  $\gamma_i$ , and this requires extra information culled from the regions adjacent to  $\gamma_i$ . Additionally, the constant value approximation yielded jumps in the error with certain discretizations; see Figure 3. Since the implementation uses a DG method with  $r = 2$ , a quadratic interpolation was chosen that uses the value of the region  $\gamma_i$  and some value from each of the regions directly adjacent, although in theory higher degree interpolations could be implemented similarly, using appropriately more points.

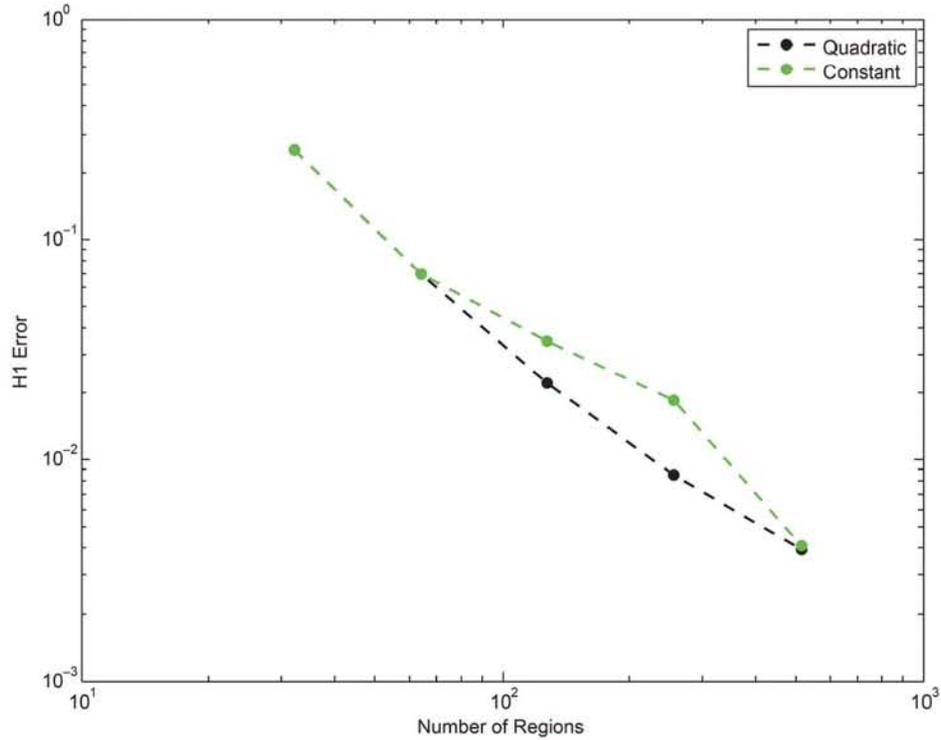


Figure 3: Comparison of the H1 error for the example described in Section 4.3.2 for both quadratic and constant interpolation from FV to DG regions. Note the spikes in the constant error for meshes of size 128 and 256 elements.

There are four different cases possible for region  $\gamma_i$  (using a notation where  $I_1^n$  and  $I_2^n$  as the defining indices used for the sectioning at time-step  $n$  and  $N$  as the number of regions in the domain):

- (i)  $(i - 1), (i + 1) \neq I_1^n, I_2^n$
- (ii)  $i = I_2^n + 1$ , or  $(i = 1$  and  $I_1^n \neq 2)$
- (iii)  $i = I_1^n - 1$ , or  $(i = N$  and  $I_2^n \neq N - 1)$
- (iv)  $(i = 1$  and  $I_1^n = 2)$ , or  $(i = N$  and  $I_2^n = N - 1)$ .

(i) When  $(i - 1), (i + 1) \neq I_1^n, I_2^n$ , this means that  $\gamma_i$  is on the interior of the set of regions that switches from an FV section to the DG one. In other words, both  $\gamma_{i-1}$  and  $\gamma_{i+1}$  are FV regions (i.e.  $\gamma_{i-1}, \gamma_{i+1} \in FV_1^n \cup FV_2^n$ ). For this case, the most natural choice of points to interpolate from are the value  $u_i^n$  at the midpoint of the FV region at time-step  $n$ , and the values at the midpoints of  $\gamma_{i-1}$  and  $\gamma_{i+1}$ , the two adjacent FV regions at time-step  $n$ :  $u_h^n|_{\gamma_{i-1}}$  and  $u_h^n|_{\gamma_{i+1}}$ , respectively.

Interpolating for the coefficients  $a_{i_j}$  yields the linear system

$$\begin{pmatrix} 1 & -2 & 4 \\ 1 & 0 & 0 \\ 1 & 2 & 4 \end{pmatrix} \begin{pmatrix} a_{i_0} \\ a_{i_1} \\ a_{i_2} \end{pmatrix} = \begin{pmatrix} u_h^n|_{\gamma_{i-1}} \\ u_h^n|_{\gamma_i} \\ u_h^n|_{\gamma_{i+1}} \end{pmatrix}$$

which can be easily solved and implemented as the matrix is invertible and well-conditioned.

(ii) When  $i = I_2^n + 1$ , this corresponds to the case where the region  $\gamma_i$  is the left-most region swapping from a FV section to the DG one; namely,  $\gamma_{i-1} \in DG^n$  and  $\gamma_{i+1} \in FV_2^n$ . In this instance, take the first interpolation point to be  $u_{i+1}^n$  from the right-hand FV region and the second as the DG value for  $y_{i-1}$  from the solution for region  $\gamma_{i-1}$ . Thus the three interpolation points are  $u_h^n|_{\gamma_i}$ ,  $u_h^n|_{\gamma_{i+1}}$ , and  $u_h^n(\bar{y}_{i-1})$ .

Interpolating for the coefficients  $a_{i_j}$  yields the linear system

$$\begin{pmatrix} 1 & -1 & 1 \\ 1 & 0 & 0 \\ 1 & 2 & 4 \end{pmatrix} \begin{pmatrix} a_{i_0} \\ a_{i_1} \\ a_{i_2} \end{pmatrix} = \begin{pmatrix} u_h^n(\bar{y}_{i-1}) \\ u_h^n|_{\gamma_i} \\ u_h^n|_{\gamma_{i+1}} \end{pmatrix}$$

which can be dealt with in an analogous way as to that in (i).

When, instead,  $i = 1$  and  $I_1^n \neq 2$ , this means that the boundary value  $\alpha$  can be used as the left interpolation point, leading to the similar linear system

$$\begin{pmatrix} 1 & -1 & 1 \\ 1 & 0 & 0 \\ 1 & 2 & 4 \end{pmatrix} \begin{pmatrix} a_{1_0} \\ a_{1_1} \\ a_{1_2} \end{pmatrix} = \begin{pmatrix} \alpha \\ u_h^n|_{\gamma_1} \\ u_h^n|_{\gamma_2} \end{pmatrix}$$

(iii) When  $i = I_1^n - 1$ , this means that the region  $\gamma_i$  is the right-most region swapping from a FV section to the DG one, or that, more precisely,  $\gamma_{i-1} \in FV_1^n$  and  $\gamma_{i+1} \in DG^n$ . Use the interpolation points  $u_h^n|_{\gamma_i}$ ,  $u_h^n|_{\gamma_{i-1}}$ , and the right endpoint value for  $y_i$  from the DG region  $\gamma_{i+1}$ ,  $u_h^n(\bar{y}_i)$ . This leads to the linear system

$$\begin{pmatrix} 1 & -2 & 4 \\ 1 & 0 & 0 \\ 1 & 1 & 1 \end{pmatrix} \begin{pmatrix} a_{i_0} \\ a_{i_1} \\ a_{i_2} \end{pmatrix} = \begin{pmatrix} u_h^n|_{\gamma_{i-1}} \\ u_h^n|_{\gamma_i} \\ u_h^n(\bar{y}_i) \end{pmatrix}$$

which is, again, easily solved. For the case when  $i = N$  and  $I_2^n \neq N - 1$ , then use  $\beta$  as the right interpolation point and get

$$\begin{pmatrix} 1 & -2 & 4 \\ 1 & 0 & 0 \\ 1 & 1 & 1 \end{pmatrix} \begin{pmatrix} a_{i_0} \\ a_{i_1} \\ a_{i_2} \end{pmatrix} = \begin{pmatrix} u_h^n|_{\gamma_{N-1}} \\ u_h^n|_{\gamma_N} \\ \beta \end{pmatrix}$$

(iv) In this final case,  $\gamma_i$  is a boundary region switching from FV to DG, and is bordered on the non-boundary side by a region in  $DG^n$ . In other words, this only occurs (in this implementation) when there is only one region in either  $FV_1$  or  $FV_2$ . Taking the interpolation points as  $u_h^n|_{\gamma_i}$  and either both  $v = \alpha$  and  $w = u_h(\bar{y}_1)$  or both  $v = u_h(\bar{y}_{N-1})$  and  $w = \beta$  leads to the linear system

$$\begin{pmatrix} 1 & -1 & 1 \\ 1 & 0 & 0 \\ 1 & 1 & 1 \end{pmatrix} \begin{pmatrix} a_{i_0} \\ a_{i_1} \\ a_{i_2} \end{pmatrix} = \begin{pmatrix} v \\ u_h^n|_{\gamma_i} \\ w \end{pmatrix}$$

### 3.4 Initial Discretization

At the initial time value,  $t_0$ , some initial partition for  $FV_1^0$ ,  $DG^0$ , and  $FV_2^0$  must be chosen. For the model problem (1), there is given an initial condition function  $u^0$ . Since no information about this initial condition can be inferred without implicitly choosing an initial partition, choose  $DG^0 = \{\gamma_i : i = 1, \dots, N\}$ ,  $FV_1^0 = FV_2^0 = \emptyset$ . This ensures that as much accuracy in the final solution is maintained as possible, and, since this costly partition used only used at the beginning time-step, the added computational cost is comparatively rather marginal and well worth the added benefit in accurately gauging an appropriate partition for the next time-step. This also helps approach problems like that depicted in Figure 5; if this precaution is not taken, the geometry of the initial condition (namely, piecewise constant) is such that an adaptive sectioning is not appropriate and will result in undue loss in accuracy.

## 4 Results

Since both the FV scheme and backward-Euler time-stepper used has a minimum first-order error bound in the  $L^2$  and  $H^1$  norms, the expected numerical convergence rates (NCR) will be bounded from below by one. Throughout, we take  $\Delta t = \frac{1}{2}\Delta x^2$  to minimize time-steppers effect on the numerical convergence rates.

### 4.1 Norms

Both the  $L^2$  norm and the  $H^1$  “energy” norm are used in the numerical results and are defined in the usual way:

$$\|v\|_{L^2} = \sqrt{\sum_{i=1}^N \int_{\gamma_i} v^2 dx} \quad \|v\|_{H^1} = \sqrt{\sum_{i=1}^N \int_{\gamma_i} (v')^2 dx + \sum_{k=1}^{MDG} \sum_{j=1}^{NDG_k} \frac{[v(x_j^{DG_k})]^2}{h}}$$

For the numerical results below, the errors are measured at the final time,  $t = t_F$ .

### 4.2 Fixed Partition Results

An for a fixed sectioning is introduced to suggest the efficacy of the developed solver. Taking exact solution  $u(x, t) = x(x - 1)e^t$  with  $x, t \in [0, 1]$ ,  $\varphi = 10$ ,  $\kappa = 2$  (Figure 4) gives the expected rates. We take a 4-section partition of the domain, i.e. 2 DG regions (in red) and two FV regions (in blue).

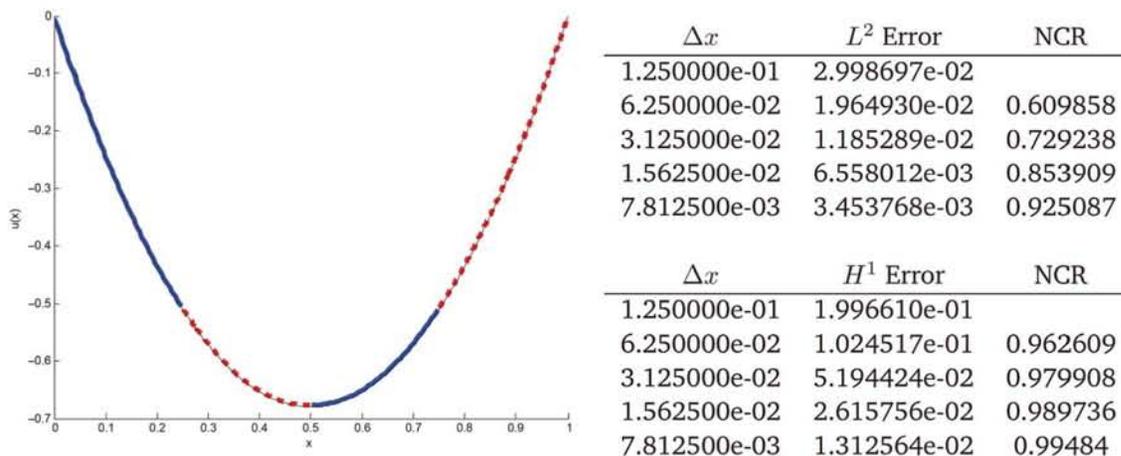


Figure 4: A fixed partition result. For distinct partitions are used, 2 DG (red) and 2 FV (blue).

### 4.3 Adaptive Partition Results

#### 4.3.1 Qualitative Results

It is straightforward to construct a set of inputs that yield a wave-like shape that traverses the domain. For instance, taking  $\kappa = 0.1$ ,  $\varphi = 100$ ,  $u(0) = u(2) = 0$ ,  $t \in [0, \frac{1}{100}]$ ,  $N = 384$ , and

$$u^0(x) = \begin{cases} 3 & \text{if } x \in [0.1, 0.2] \\ 1 & \text{if } x \in [0.3, 0.4] \\ 0 & \text{else} \end{cases}$$

and running the region-swapping solver for  $\Delta t = \frac{1}{128^2} = \frac{1}{16384}$  yields the three plots in Figure 5 at  $t = 0, \frac{1}{200}$ , and  $\frac{1}{100}$ .

Qualitatively, the sole FV solution at  $t = t_F$  is more diffused; it shares the vague outline of the sole DG solution, but they are distinctly different solutions. The region-swapping solution, however, is, to the resolution of the plot, almost identical to the sole DG solution, the only discernible differences between the two occurring at the interface points. The number of DG regions is also significantly curtailed with the region-swapping method: the average number of DG regions at is just under 73 at a given time-step, and the range varied (excluding the entirely DG initial condition) from 45 to 82, overall a significant drop from the 384 of the sole DG method. Table 1 tabulates the size of the resulting systems for the region-swapping scheme; the size of the resulting matrix never exceeds half that of the system for the sole DG implementation. At least for this specific example, it seems that the desire for a solution comparable to the sole DG solution at a significantly lower computational cost has been met.

| t     | Size | Ratio  |
|-------|------|--------|
| 1/200 | 526  | 0.4566 |
| 1/100 | 544  | 0.4722 |

Table 1: Size of linear system at time t for example shown in Figure 5; also includes ratio of region-swapping system size to that for the sole DG implementation

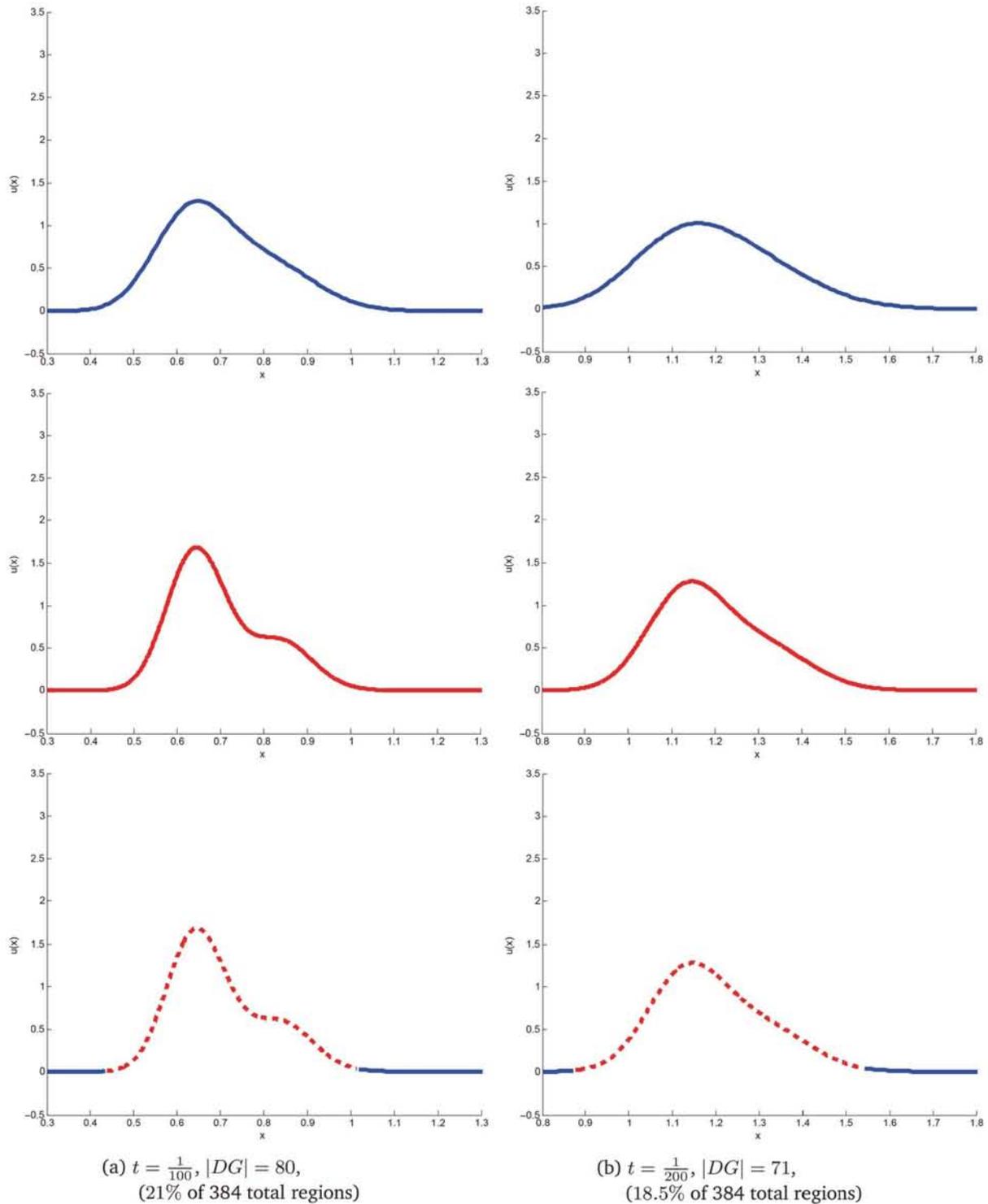


Figure 5: Two plots at different time values, each with three components: a plot of the sole FV solution (top), one of the sole DG solution (middle), and one of the region-swapping solution (bottom). In all plots, blue denotes FV solutions and red DG solutions. The solution traverses the domain, but the x-axis have been cut to an appropriate subset of the domain of width 1 for clarity. Note that the region-swapping and sole DG solution are nearly identical to the resolution of the plot, while the sole FV solution is significantly more diffused than both after the initial time  $t_0 = 0$ .

### 4.3.2 Quantitative Results

For the region-swapping scheme to be considered a robust method for solving the model PDE (1), the same numerical convergence rates are expected as for a sole DG or FV implementation. Since the backward-Euler time-stepper used for the scheme is first order, as is the FV method, it is expected that a properly working region-swapping method will yield numerical convergence rates of 1 in both the  $L^2$  and energy norms.

To simulate wave-like behavior with an elementary function, the arctan function is chosen, the argument being both spatially and time dependent:  $u(x, t) = -\arctan(10(x-t)-5) + \frac{\pi}{2}$ . The exact boundary conditions  $u(0)$  and  $u(4)$  are used, and the values  $\varphi = 2, \kappa = 0.4, t_0 = 0$ , and  $t_F = 1$  are taken.

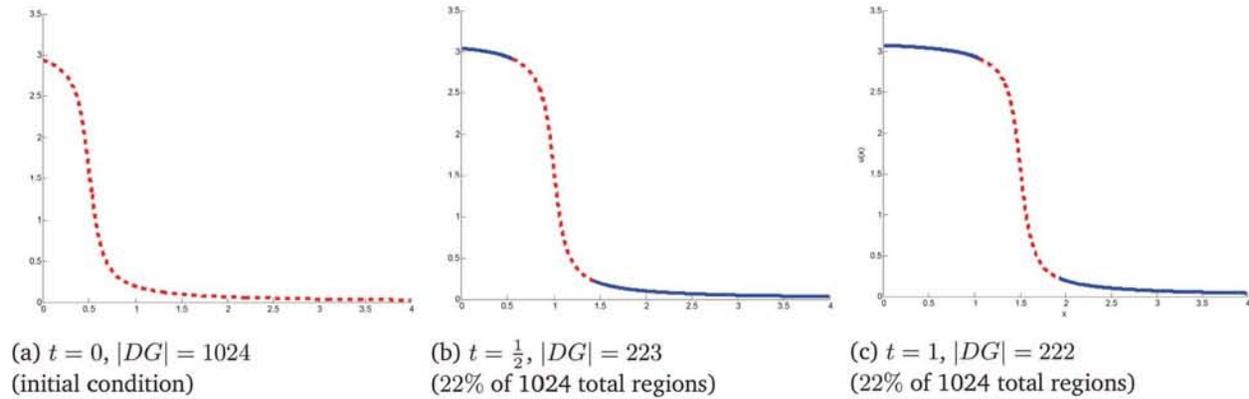


Figure 6:

$$u_t - 0.4u_{xx} + 2u_x = f$$

$$x \in [0, 4], t \in [0, 1]$$

$$\text{exact solution: } u(x, t) = -\arctan(10(x-t)-5) + \frac{\pi}{2}$$

boundary conditions and initial condition derived from exact solution.

Three plots of the region-swapping scheme solution and the exact solution with  $N = 1024$ ; the two are indistinguishable. Below, charts of numerical convergence rates in the  $L^2$  and  $H^1$  norm errors.

| $\Delta x$   | $L^2$ Error  | NCR      | $\Delta x$   | $H^1$ Error  | NCR     |
|--------------|--------------|----------|--------------|--------------|---------|
| 1.250000e-01 | 4.211964e-02 |          | 1.250000e-01 | 2.539514e-01 |         |
| 6.250000e-02 | 1.852333e-02 | 1.18515  | 6.250000e-02 | 6.983134e-02 | 1.86261 |
| 3.125000e-02 | 9.109862e-03 | 1.02384  | 3.125000e-02 | 2.211028e-02 | 1.65916 |
| 1.562500e-02 | 4.561075e-03 | 0.998055 | 1.562500e-02 | 8.598715e-03 | 1.36252 |
| 7.812500e-03 | 2.288789e-03 | 0.99479  | 7.812500e-03 | 3.913292e-03 | 1.13574 |

The numerical convergence rates are just as expected, approaching 1 in the  $L^2$  and  $H^1$  norm as the grid refinement is doubled for each iteration. Also notable are the size of the resulting systems for the region-swapping method, as tabulated in Table 2; the size hovers around 20% of that for the sole DG scheme.

| t   | Size | Ratio  |
|-----|------|--------|
| 1/2 | 223  | 0.2177 |
| 1   | 222  | 0.2168 |

Table 2: Size of linear system at time t for example shown in Figure 6; also includes ratio of region-swapping system size to that for the sole DG implementation

#### 4.4 Advantages of Region-Swapping

The desired result of the region-swapping scheme laid out above is the production of an approximate solution of accuracy sufficiently close to that of the sole DG implementation with an appreciably smaller cost. The computational cost of the method will scale with the size of the linear system solved at each iteration (i.e. the degrees of freedom); as such, we will measure the cost of the methods by the size of the resulting linear system. For a mesh of  $N$  elements, the size of the FV linear system will be  $N$ , the size of the DG linear system of degree  $r$  will be  $(r+1)N$ , and the size of the region-swapping linear system will be  $(r+1)dN + (1-d)N = (rd+1)N$ , where  $0 \leq d \leq 1$  is the ratio of elements in the domain which are contained in a DG section at a given time.

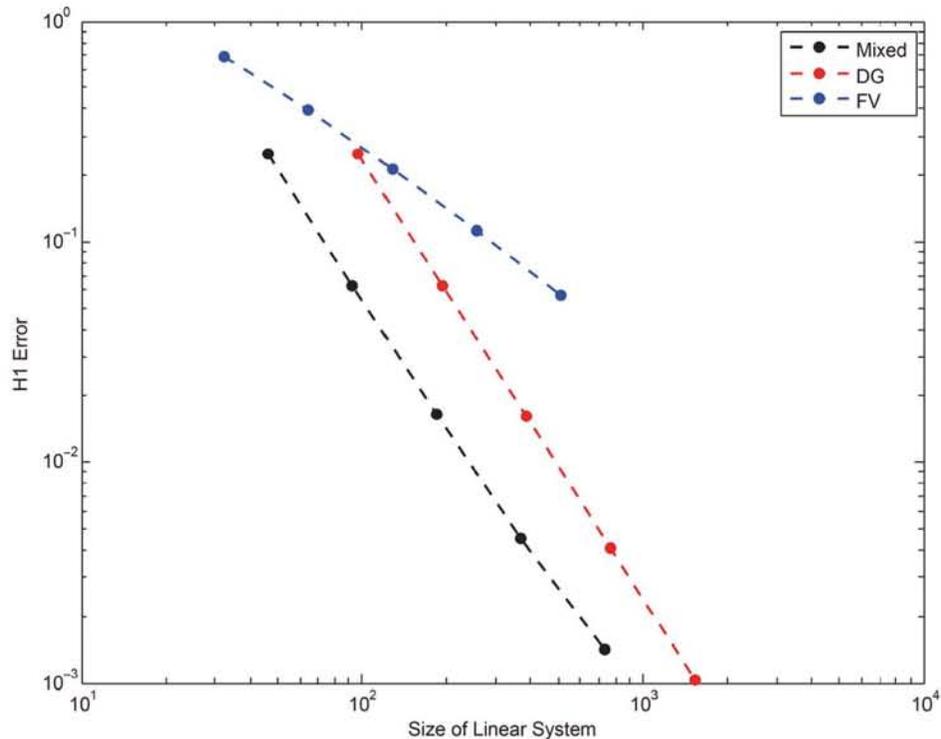


Figure 7: Plot of H1 error on the final DG section of the region-swapping solution. Also included are the errors over the same section for the sole DG and FV solutions. For a given system size, the mixed region-swapping solution yields the smallest error.

Consider the example in Section 4.3.2, and consider the error over the portion of the domain which is DG regions in the region-swapping solution. This approach is taken to test the efficacy of the region-swapping solver over only that portion of the domain where it is reasonable to expect convergence similar to the sole DG solver (namely, those DG regions at the final time-step). The plot in Figure 7 shows the H1 error over the DG regions of various approximate solutions as a function of their producing size of linear system. For a given system size, the mixed region-swapping solution yields the lowest error. Additionally, the interpolating lines for the mixed and the DG errors are of approximately the same slope, implying a comparable convergence rate. Thus, the region-swapping method yields an approximate solution with comparable accuracy to the DG solution with a lower computational cost.

## 5 Conclusion

A coupled DG/FV solver for a parabolic convection-diffusion partial differential equation with Dirichlet boundary conditions in one dimension was developed. The solver accepts an arbitrary discretization of the domain into DG and FV sections, and preserves the numerical convergence rates of the appropriate sole solver. A region-swapping method was developed that would reassign the sectioning at each time-step so that the higher-order DG method is used when great accuracy is desired and the lower-cost FV method is used otherwise. The region-swapping method produces an approximate solution sufficiently close to the sole DG solution but with a significantly smaller linear system that is proportionally cheaper to solve as the grid discretization is refined. Most importantly, the numerical convergence rates for the region-swapping method are conserved.

This particular avenue of pursuit is rife with opportunity for future work. The work [1] establishes the convergence, existence, and uniqueness of the solution for a diffusion equation with two sections, as well as an error bound, and the argument for the same for an arbitrary sectioning follows similarly, but that for a parabolic convection-diffusion equation in general is a nontrivial extension that should be established for a robust scheme. Most pressingly, the theoretical error bounds for the region-swapping scheme have not been established. Extensions of the method to higher dimensions are also desirable to properly address the motivating problem, as well as a robust accompanying solver in a universal and portable programming language such as C or C++.

## References

- [1] Y. Sun. "A coupled multinumeric method for elliptic problems". Rice University. 2009.
- [2] B. Riviere. "Discontinuous Galerkin Methods for Solving Elliptic and Parabolic Equations: Theory and Implementation". SIAM. 2008.
- [3] R. Eymard, T. Gallouet, R. Herbin. "Finite Volume Methods". October 2006.
- [4] M. Heath. "Scientific Computing: An Introductory Survey". McGraw Hill. 2005.
- [5] P. Chidyagwai, I. Mishev and B. Riviere. "On the coupling of finite volume and discontinuous Galerkin method for elliptic problems". **Journal of Computational and Applied Mathematics**, 231 p.2193-2204, 2011, doi:10.1016/j.cam.2010.10.017.
- [6] B. Riviere, P. Chidyagwai, I. Mishev. "On the coupling of finite volume and discontinuous Galerkin for reservoir simulation problems". Proceedings of the SPE conference. 2011.
- [7] B. van Leer. "Upwind and High-Resolution Methods for Compressible Flow: From Donor Cell to Residual-Distribution Schemes". **Communications in Computational Physics**, Vol. 1, No. 2, pp. 192-206. 2006.