Solving Nonlinear Advection-Diffusion Problems Using a Second Order Accurate Finite Volume Method

By Zachary C. Cordero

Abstract: A second order accurate finite volume method for solving nonlinear advection-diffusion problems is explained, validated against a linear test problem with an analytical solution, and implemented in several non-linear case studies. Both 1 and 2 dimensional problems are studied to demonstrate the ease with which the method is extended to higher dimensions.

I. Introduction

When medication is injected into a patient's vein, the medication is carried or advected in the direction of the blood flowing through the vein. Additionally, the medication, which is initially concentrated, spreads out over time due to diffusion. Assuming advection and diffusion are the only two processes occurring, the medication's concentration profile, $c(x,t)$, evolves in time according to the following PDE:

$$\partial_t c = \text{div}\,\{h(c)\nabla V(x) + \nabla r(c)\} \quad (1)$$

where the first and second terms in the curly brackets describe transport due to advection and diffusion, respectively.

In many transport problems, (1) is linearized by setting $h(c)$ and $r(c)$ equal to c. When this is done, we can use standard explicit finite difference techniques to evolve a starting concentration profile in time. However, transport problems in which $h(c)$ and $r(c)$ are more complex functions of c also arise. In such problems, (1) becomes nonlinear and explicit finite difference techniques do not suffice because it is no longer possible to take derivatives of the discontinuities that emerge during the evolution of such problems.

In this paper, a second order accurate, finite volume scheme for solving nonlinear advection-diffusion problems [1] is implemented, validated, and applied in some 1- and 2-D nonlinear, advection-diffusion case studies. We selected a finite volume method because such methods are easy to implement, straightforward to extend to higher dimensions on a Cartesian grid, and converge to a physically correct weak solution when applied to nonlinear problems, provided certain conditions are met.

The structure of this paper is as follows. We begin in section 2 by explaining the mathematical underpinnings of finite volume methods and then explain the technique utilized in this manuscript. This includes an explanation of how using a linear interpolation scheme with a slope limiter can lead to second order accuracy in space while giving solutions free of unphysical oscillations. In section 3, the method and expected order of accuracy are validated using a simple linear test case. Finally, in section 4, the method is applied to a non-linear diffusion problem: fluid flow in a

porous medium where the function $r(c)$ is of the form c^m with m equal to some integer greater than 1.

II. Finite Volume Methods

1. *Background*

Eqn. 1 is a microscopic mass-balance law: the change in concentration in a differential volume element is equal to the divergence of the flux across that volume element. As alluded to above, issues arise when this differential picture is applied to nonlinear problems. This is because in non-linear problems, characteristics tracking the evolution of the initial conditions can intersect to form discontinuities and the derivatives in (1) are not defined at these discontinuities. Therefore, a better way to handle such non-linear problems is to avoid taking derivatives by reformulating the PDE as an integral. Considering a discrete interval in space-time, the integral form of (1) can be written in 1-dimension as

$$\int_{x_1}^{x_2} c(x,t_2)dx = \int_{x_1}^{x_2} c(x,t_1)dx + \left(\int_{t_1}^{t_2} f(x_1,t)dt - \int_{t_1}^{t_2} f(x_2,t)dt\right) \quad (2)$$

where

$$f(x,c) = h(c)\nabla V(x) + \nabla r(c). \quad (3)$$

In words, (2) says that the change in the mean value of the concentration across the interval $(x_2 - x_1)$ in space is equal to the total flux across the boundaries to that interval between times t_1 and t_2. A discrete formulation of (2) that forms the basis of most finite volume numerical methods is

$$c_j^{n+1} = c_j^n - \frac{\Delta t}{\Delta x}(\tilde{F}_{j+1/2} - \tilde{F}_{j-1/2}) \quad (4)$$

where c_j^n is the mean concentration of cell j at time step n and $\tilde{F}_{j+1/2}$is the total flux across the boundary between the cells at $j + \frac{1}{2}$ during 1 time step. In (4) we are stepping forward in time using Forward Euler.

Eqns. (2) and (4) are called the continuous and discrete conservation forms of the original PDE because they guarantee that total concentration is conserved. That this is so for Eqn. 4 can be seen by first discretizing the entire domain into a series of adjacent cells and then summing each of their respective Eqn. 4's. When this is done, the flux terms of neighboring cells cancel out and we obtain

$$\sum_{n=A}^{B} c_j^{n+1} = \sum_{n=A}^{B} c_j^n - \frac{\Delta t}{\Delta x}(\tilde{F}_{B+1/2} - \tilde{F}_{A-1/2}) \quad (6)$$

where all that remains are fluxes at the domain boundaries A and B. Thus, net changes in total concentration are only due to mass entering or exiting at these boundaries. In addition to avoiding derivatives of discontinuous functions, there is a second benefit to rewriting (1) in a conservation form and using it to solve nonlinear PDEs numerically: according to the Lax-Wendroff theorem, if a conservative and consistent finite volume method converges, then it will converge to a weak solution and shocks or discontinuities will be in a correct position (though not necessarily the physically correct position) [2].

Whether a consistent and conservative finite volume method converges is another issue that is dictated by how the numerical approximation's total variation evolves over time. The total variation (TV) is a measure of how much the approximation oscillates and is defined as

$$TV = \sum_{j=A}^{B} |c_{j+1} - c_j|. \quad (7)$$

A consistent and conservative method will converge provided the method is TV stable, meaning that $TV(n\Delta t)$ is bounded (the Lax-Friederichs method that we implement ensures this by actually being TV-diminishing).

Finally, to guarantee that the method converges to the physically relevant weak solution, the method must also meet some form of an entropy condition, which is the CSE equivalent to the 2nd law thermodynamics: global entropy can only increase. To a computational scientist, this translates into the idea that information about the initial data can only disappear (i.e., shocks can only absorb characteristics, not emit them). In practice, the methods accomplish this by incorporating some numerical dissipation, which serves to "diffuse" away some of the initial information at shocks or rarefactions. In summary, a consistent and conservative finite volume scheme will converge to the physically relevant weak solution provided it is TV-stable and satisfies an entropy condition. The challenge in coming up with accurate finite volume schemes lies in developing methods for estimating the flux with just enough numerical dissipation.

We now turn to the practical matter of relating (1) to (4). Because (1) is an advection-diffusion equation, it is a parabolic PDE. However, finite volume numerical methods were originally developed for solving nonlinear hyperbolic equations. This motivates recasting (1) in the form of the standard 1-dimensional hyperbolic equation by writing the terms in the curly brackets as

$$h(c)\partial_x V(x) + \partial_x r(c) = h(c)\partial_x\big(V(x) + s(c)\big) = h(c)\partial_x(A(x,c)) \quad (8)$$

where

$$s(c) = \int_1^c \frac{r'(c^*)}{h(c^*)}\, dc^*. \quad (9)$$

We treat the function A as a velocity and approximate it at each boundary using centered differences:

$$A_{j+\frac{1}{2}} = -\left[\frac{V(x_{i+1}) - V(x_i)}{\Delta x} + \frac{s(c_{i+1}) - s(c_i)}{\Delta x}\right]. \quad (10)$$

2. *First Order Methods*

Given (4), the main challenge in developing a finite volume scheme is finding an accurate approximation to the flux, F. Most of the first order accurate methods for estimating the flux involve the same procedure: first discretize the solution at time step n into a series of neighboring cells with piecewise constant concentrations equal to the mean concentration of that cell, and then estimate the flux using some combination of the mean concentration in neighboring cells. For example, the local Lax-Friederichs (LLxF) method is a centered scheme that estimates the flux as follows:

$$\tilde{F}_{j-1/2} = \frac{1}{2}\left[f(c_{j-1}) + f(c_j) - \alpha_{j-1/2}(c_j - c_{j-1})\right] \quad (12)$$

where

$$\alpha_{j-1/2} = \max\left(|f'(c_{j-1})|, |f'(c_j)|\right). \quad (13)$$

Approximating a to be a constant and inserting (12) into (4) gives

$$U_j^{n+1} = U_j^n - \frac{\Delta t}{2\Delta x}(f(c_{j+1}) - f(c_{j-1}) - \alpha(c_{j+1} - 2c_j + c_{j-1})) \quad (14)$$

Inspection of (14) reveals a discretized second derivative term with a prefactor α that results from the second term in (12). This second derivative contributes some numerical dissipation to the method and the prefactor α controls the magnitude of the dissipation. Though this numerical dissipation is what enables the LLxF method to satisfy the entropy condition, it is problematic in that it does not disappear when it is unnecessary, i.e., when the solution is smooth. As a result, shocks and even smooth regions in solutions calculated using first order methods tend to become rounded off as the system evolves. This tendency to smoothen out is illustrated in Figures 1a-d, which compare solutions to a linear advection problem with a constant speed of 1, periodic boundary conditions, and an initial condition consisting of a smooth Gaussian and a box function calculated using the first order accurate Lax-Friederichs and upwind schemes. As is apparent from the figure, both schemes' numerical dissipations cause the initial features in the simulation to become rounded over time.

For the linear advection problem just discussed, a logical next choice for trying to improve the accuracy would be to calculate the flux using a method that is second order accurate in space such as the Lax-Wendroff method:

$$\tilde{F}_{j-1/2} = \frac{1}{2}A\left(c_j + c_{j-1}\right) - \frac{1}{2}\frac{\Delta t}{\Delta x}A^2(c_j - c_{j-1}). \quad (15)$$

where A is a speed that is set equal to 1 for this test problem. In essence, the Lax-Wendroff scheme is a piecewise linear interpolative scheme that uses the upwind slope to calculate the slope of the approximate function for c in each cell. Encouragingly, Figures 1e and f demonstrate how the Lax-Wendroff method accurately evolves the smooth Gaussian in time for the linear advection equation. However, the box develops large, unphysical oscillations at its leading and trailing edges. These oscillations arise because the dominant error in the Lax-Wendroff method is dispersive in nature (i.e., it features an odd derivative of the solution). Consequently, the various modes that contribute to the shape of the box travel at different speeds and thus disperse over time.

3. *Second Order Methods*

In light of the results just described, an even better next step for achieving second order accuracy and minimal dissipation while avoiding spurious oscillations at discontinuities is to generate a hybrid method that rapidly shifts from second order accurate in smooth regions to first order accurate and dissipative around discontinuities. This is the idea behind the high-resolution, 2nd order accurate

schemes incorporating slope limiters. In these schemes, the individual cells are approximated as being piecewise linear functions,

$$c^n(x,t_n) = c_j^n + \sigma_j^n(x - x_j), \tag{16}$$

and the key to these methods is designing a function that calculates the slope, σ_j^n, while accounting for whether the approximation at *j* is locally smooth, at a discontinuity, or at an extremum. When the solution is smooth, the slope should reflect the concentration of the neighboring cells, i.e., behave more like Lax-Wendroff. Conversely, near discontinuities or extrema the slope should go to 0 so that the approximation to the concentration in that cell becomes piecewise constant, thereby increasing the local numerical dissipation and causing the flux function to behave more like the standard LLxF.

The first step in developing such a function is to select an indicator of whether a point is at a discontinuity or an extrema. A simple indicator is the ratio of slopes around a point

$$\theta_j = \frac{c_j - c_{j-1}}{c_{j+1} - c_j}. \tag{17}$$

If θ_j is close to 1, then the function is varying smoothly around *j*; if it is negative, then *j* is an extrema; and if it is very large or small, then *j* is to the right or left of a discontinuity, respectively. Using θ_j as an indicator of a given cell's local environment, one can then insert it into a slope limiter that modifies the slope to account for this environment

$$\phi_j = \frac{\theta + |\theta|}{1 + |\theta|}. \tag{18}$$

This ϕ is called the van Leer slope limiter and its behavior as a function of θ is shown in Figure 2. Also shown are two other slope limiters (Superbee and minmod) that provide upper and lower bounds to the range of allowable values for slope limiter functions while still guaranteeing the overall method is TVD. We incorporate this function into the interpolation scheme by using it to calculate the values of *c* at either side of a cell as follows

$$\begin{aligned} c_{j+\frac{1}{2},-} &= c_j + \frac{1}{2}\phi_j(\theta_j)(c_{j+1} - c_j) \\ c_{j-\frac{1}{2},+} &= c_j - \frac{1}{2}\phi_j(\theta_j)(c_{j+1} - c_j) \end{aligned} \tag{19}$$

Finally, these *c* values are used to calculate a LLxF flux at the cell boundary:

$$\tilde{F}_{j+\frac{1}{2}} = \frac{A_{j+\frac{1}{2}}}{2}\left(f(c_{j+\frac{1}{2},-}) + f(c_{j+\frac{1}{2},+})\right) - \frac{|A_{j+\frac{1}{2}}|\alpha_{j+\frac{1}{2}}}{2}\left(c_{j+\frac{1}{2},+} - c_{j+\frac{1}{2},-}\right). \tag{20}$$

Inspection of (20) reveals how the magnitude of the numerical dissipation now depends on the difference in interpolated *c* values immediately to the left and right of the cell boundary; when this difference is 0, (i.e., the approximation is a straight line) the dissipation disappears.

Figure 3 demonstrates that ϕ behaves as we would hope. It approaches 0 (and consequently *F* approaches the standard LLxF formula) as θ approaches 0 and is 0 for all negative θ. It is 1 when θ is 1 (and as a result, the numerical dissipation

disappears). And it has a maximum value of 2 as θ blows up so that it can accurately capture discontinuities without leading to non-TVD behavior. Additionally, the linear interpolation diagrams in Figure 4 confirm that this van Leer limiter is also at least TV-stable because at no point is the magnitude of an edge value of *c* greater than the maximum magnitude of the mean values of *c* in the domain. Figures 5a-c demonstrate how incorporating the slope limiter restricts the numerical dissipation to regions close to discontinuities.

In summary, combining the ideas developed in the past three sections, we arrive at a second order accurate method for calculating the numerical flux in a non-linear advection-diffusion problem:

$$\tilde{F}_{j+\frac{1}{2}} = \frac{A_{j+\frac{1}{2}}}{2}\left(f(c_{j+\frac{1}{2},-}) + f(c_{j+\frac{1}{2},+})\right) - \frac{|A_{j+\frac{1}{2}}|\alpha_{j+\frac{1}{2}}}{2}\left(c_{j+\frac{1}{2},+} - c_{j+\frac{1}{2},-}\right),$$

$$A_{j+\frac{1}{2}} = -dV_{j+\frac{1}{2}} - dh(c)_{j+\frac{1}{2}}$$

$$\alpha_{j+\frac{1}{2}} = \max\left(f'(c)\right) \text{ over all } u \text{ between } c_j \text{ and } c_{j+1} \qquad (21)$$

$$c_{j+\frac{1}{2},-} = c_j + \frac{1}{2}\phi_j(\theta_j)(c_{j+1} - c_j)$$

$$c_{j+\frac{1}{2},+} = c_{j+1} - \frac{1}{2}\phi_j(\theta_{j+1})(c_{j+2} - c_{j+1}).$$

Using this flux approximation with (4) gives the complete numerical scheme. Figures 6 shows results from a simulation similar to those in Figure 2 calculated using the second order accurate given by (21). Note how much sharper the discontinuities are in the simulations using (21).

The time step constraint for this method is like the CFL time step constraint of other explicit diffusion methods. For example, the time step constraint is $\Delta t \leq \Delta x^2/2$ for a 1-D linear advection-diffusion problem solved using an explicit second order accurate centered difference scheme in space and a forward Euler method in time. Similarly, the time step constraint for the scheme in (21) is

$$\Delta t \max_j \left|V(x_{j+1}) - V(x_j) + s(c_{j+1}) - s(c_j)\right| \leq \frac{\Delta x^2}{2}. \qquad (22)$$

Adapting the scheme to higher dimensions on a Cartesian grid is straightforward. One simply uses dimensional splitting whereby the problem is broken up into a series of 1-dimensional problems along each axis. For example, to move forward 1 time step in a 2-dimensional problem, we first update the concentrations along the y-direction and then the x-direction as follows:

$$c^*_{i,j} = c^n_{i,j} - \frac{\Delta t}{\Delta x}(\tilde{F}_{i,j+1/2} - \tilde{F}_{i,j-1/2})$$

$$c^{n+1}_{i,j} = c^*_{i,j} - \frac{\Delta t}{\Delta x}(F^*_{i+1/2,j} - F^*_{i-1/2,j}) \qquad (23)$$

Generally, some splitting error arises because the two operations do not necessarily commute, but the method is used without major error in the problems discussed next.

III. Validation

To validate the scheme described above, we first used it to study the evolution of a linear advection-diffusion problem in which

$$\begin{aligned} h(c) &= c \\ \nabla V(x) &= 1 \\ \nabla r(c) &= c \end{aligned} \tag{24}$$

The goal of these simulations was to demonstrate that the method gives results consistent with an analytical solution to the linear problem, has a spatial accuracy of $O(\Delta x^2)$, and can be extended to 2-dimensions on a Cartesian grid using dimensional splitting.

The Green's function of the linear advection-diffusion problem on an infinite domain is a Gaussian that translates and broadens with time according to

$$c(x,t) = \frac{1}{\sqrt{4\pi t}} e^{-(x-t)^2/4t}. \tag{25}$$

Similarly, in 2-dimensions, it can be shown using separation of variables that the Green's function is a product of Gaussians in the x and y-directions:

$$c(x,y,t) = \left(\frac{1}{\sqrt{4\pi t}}\right)^2 e^{-(x-t)^2/4t} e^{-(y-t)^2/4t} \tag{26}$$

where it is assumed that the advection velocity is 1 in both directions. These analytical solutions provide references with which to check the accuracy of our scheme by measuring the L^1 and L^{inf} error as the mesh size is refined.

To demonstrate how this error analysis is performed, we begin by using it on the explicit centered finite difference method for solving the advection-diffusion equation. In 1-D, this scheme is

$$c_j^{n+1} = c_j^n + \frac{\Delta t}{2\Delta x}(c_{j+1}^n - c_{j-1}^n) + \frac{\Delta t}{\Delta x^2}(c_{j-1}^n - 2c_j^n + c_{j+1}^n) \tag{27}$$

where we are using the Forward Euler method to evolve the system in time. Like our LLxF method, this method also has $O(\Delta x^2)$ accuracy in space, first order accuracy in time, and a time step constraint of $\Delta t \sim \Delta x^2/2$.

Simulations were performed using (5) with the initial data

$$c(x,0) = \frac{1}{\sqrt{4\pi t}} \exp\left(-\frac{x^2}{0.4}\right), \quad x \in (-5,5) \tag{28}$$

The simulations were run to T = 0.8 using 16, 32, 64, 128, and 256 grid points. The time step for each of the simulations was calculated using $\Delta t = \frac{\Delta x^2}{2}$. The boundary conditions were dealt with by updating the boundary values at each time step with the analytical solution. Figures 6a-c contain snapshots of the N = 128 simulation overlaid on the analytical solution at different times; agreement between the simulation and analytical results is visibly good. Figure 6j contains a log-log plot of Δx versus L^{inf} confirming that the method's spatial accuracy is $O(\Delta x^2)$. The other images in Figure 6 illustrate how the same set of experiments in 2-dimensions are also second order accurate. These simulations were evaluated using the 2-dimensional version of (27):

$$c_{j,k}^{n+1} = c_j^n + \frac{\Delta t}{2\Delta x}(c_{j,k+1}^n + c_{j+1,k}^n - (c_{j-1,k}^n + c_{j,k-1}^n)) + \cdots$$

$$\ldots \frac{\Delta t}{\Delta x^2}\left(c_{j-1,k}^n + c_{j,k-1}^n + c_{j+1,k}^n + c_{j,k+1}^n - 4c_{j,k}^n\right) \quad (29)$$

where k is the index along the y-direction and the x and y intervals are equal, and an initial condition of

$$c(x,0) = \left(\frac{1}{\sqrt{4\pi t}}\right)^2 \exp\left(-\frac{x^2}{0.4}\right)\exp\left(-\frac{y^2}{0.4}\right), \quad x \in (-5,5),\ y \in (-5,5). \quad (30)$$

Figure 7 are results from the same set of 1 and 2-D experiments applied to our LLxF finite volume scheme. Given (24), $s(c)$ is ln (c) and dV is 1 in both the x and y directions. A tolerance of 1e-12 was added to c when calculating p to avoid issues with taking the log of 0. The snapshots of the 1- and 2-D simulations shown in Figure 7 illustrate how the scheme was well-behaved in both dimensions. Figures 7a-c illustrate how agreement with the analytical solution is very good. The log-log plot shown in Figure 7j confirms that the simulations had a spatial accuracy of at least O(Δx^2) in both dimensions. Table 1 contains the L^1 and L^{inf} errors and the order of accuracy for all of the simulations.

IV. Flow in a Porous Medium

Having validated the spatial accuracy of the finite volume method against a model linear advection-diffusion problem with an analytical solution, we next applied it to a non-linear diffusion problem: liquid flowing under its own weight through a porous medium. Mathematically, the mass balance equation for such a problem can be expressed as

$$\partial_t c = \text{div}\left\{h(c)\nabla V(x) + \nabla c^m\right\} \quad (31)$$

where m is some integer greater than 1. Unless otherwise specified, we set $h(c)$ equal to 0 so that only diffusive-like flow is occurring.

In 2-dimenions, it is possible to derive the following weak solution to (31) on an infinite domain when m equals 2 using conservation of mass, radial symmetry, continuity of flux, and self-similarity:

$$c(x,t) = \begin{cases} \frac{M}{4\sqrt{Mt}}\left(\frac{2}{\sqrt{\pi}} - \frac{r^2}{\sqrt{4Mt}}\right) & 0 < r < r_0 \\ 0 & r_0 < r \end{cases} \quad (32)$$

where M is the total mass in the initial condition, r is the radius from the origin, and

$$r_0 = \left(\frac{16Mt}{\pi}\right)^{1/4} \quad (33)$$

This solution provides three metrics with which to gauge our simulation. First, the radius of the liquid slug should increase like $t^{1/4}$. Second, there should be a sharp interface between the liquid-bearing and liquid-free regions (this interface becomes sharper as m increases). Third, and most obviously, the concentration should always be positive; negative concentrations are unphysical.

We performed simulations using (21) with h(c) set equal to 0 and several different m (m = 2, 4, and 6) on Ω = (-5,5) x (-5,5). The initial conditions for these simulations were

$$c(x,0) = \begin{cases} 1 & 0 < r < 1 \\ 0 & 1 < r \end{cases} \quad . \tag{34}$$

The simulations ran to a final time of 5 with a time step calculated at each iteration using (22) on a 200 x 200 grid. The *s(c)* for a given m is

$$s(c) = \begin{cases} 2(U-1) & m = 2 \\ \frac{m}{m-1}(c^{m-1} - 1) & m > 2 \end{cases} \tag{35}$$

Figure 8 contains surface plots of the various simulations taken at the intervals indicated. Cross-sections through the y-z plane of the simulations are shown in the bottom row of Figure 8. In these cross-section plots, note the sharp interface between the "wet" and "dry" regions, the lack of oscillations at the interface, and the positivity of the solution, all of which agree with our expectations. Moreover, in Figure 9, log(t) is plotted versus log(r_0) for the m = 2 simulation and a linear fit to the data has a slope of 0.22, which differs only slightly from the expected value of 0.25 due to the initial conditions.

As a final demonstration of the generalizability of this method, we performed an additional 2-D simulation with a more general advection-diffusion equation:

$$\partial_t c = \text{div}\{xu + \nabla u^4\} \tag{36}$$

using a 200x200 grid and an initial condition of:

$$c(x,y,0) = \begin{cases} \exp\left(-\frac{1}{6-(x-2)^2-(y+2)^2}\right) & \text{if } (x-2)^2 + (y+2)^2 < 6 \\ \exp\left(-\frac{1}{6-(x+2)^2-(y-2)^2}\right) & \text{if } (x+2)^2 + (y-2)^2 < 6 \\ 0 & \text{otherwise} \end{cases} \tag{37}$$

on the domain (-10, 10) x (-10, 10). As before, the time step was held constant at 1e-4. Surface plots of the results from this simulation are shown in Figure 10.

IV. Conclusions

The goal of this work was to implement the second order accurate finite volume scheme reported in [1] and to use this scheme to study some advection-diffusion problems. To this end, the second order accuracy of this scheme in space was first validated by using it to study linear advection-diffusion problems in both 1 and 2 dimensions. The method was then applied to a non-linear diffusion problem: fluid flow through a porous medium in 2 dimensions. In this simulation, the flexibility and generality of the method were emphasized by incorporating a quadratic, Burgers-type convection term. The results from the simulations described above are encouraging and having implemented this numerical method, I now plan to apply it to some more interesting problems in materials science. In particular, I would like to use it to study the microstructures that evolve during the irradiation of positive heat of mixing binary alloys. In these materials, there is a competition between radiation-induced ballistic mixing and thermally activated segregation. Both phenomena are essentially non-linear diffusion processes and so they should be amenable to study using this method.

References:

1. M. Bessemoulin-Chatard and F. Filbet, SIAM J. Sci. Comput. **34**, B559 (2012).
2. R. J. LeVeque, *Finite Volume Methods for Hyperbolic Problems* (Cambridge university press, 2002).

Figure 1 – Images taken from simulations of linear advection problems calculated using the methods indicated. The velocity in these simulations was 1, the number of gridpoints was 200, and the time step was 3e-4. Notice how the first order accurate Lax-Friederichs and upwind methods' numerical dissipations cause the shocks to disappear as the system evolves, whereas the error in the Lax-Wendroff method gives rise to spurious oscillations.

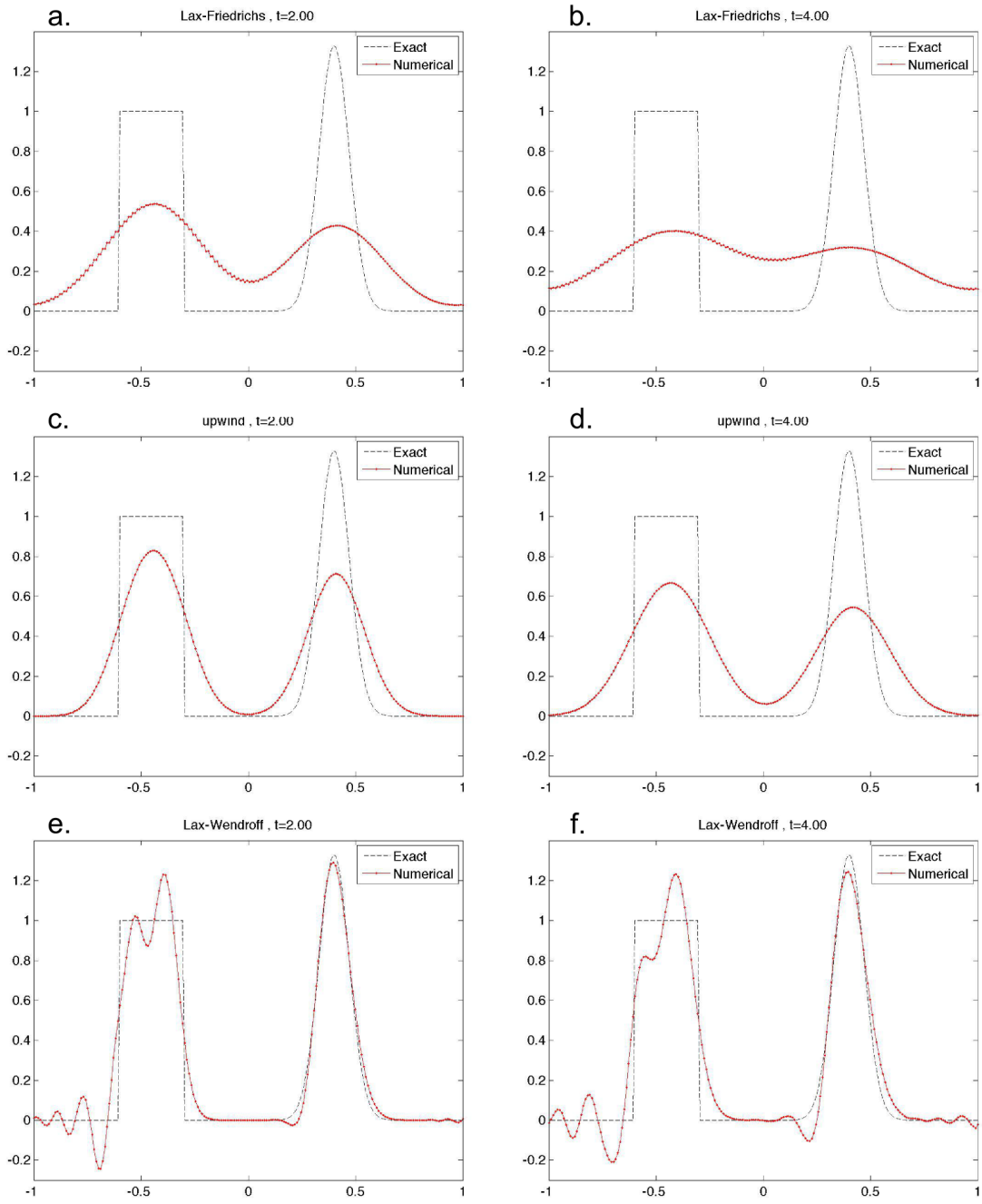

Figure 2 – Behavior of three important slope limiters (Superbee, Minmod, and van Leer) as a function of the ratio of slopes around a cell *j*. The region between the Superbee and Minmod slope limiters contains second order accurate and TVD slope limiters. The van Leer limiter was the limiter used in the present work.

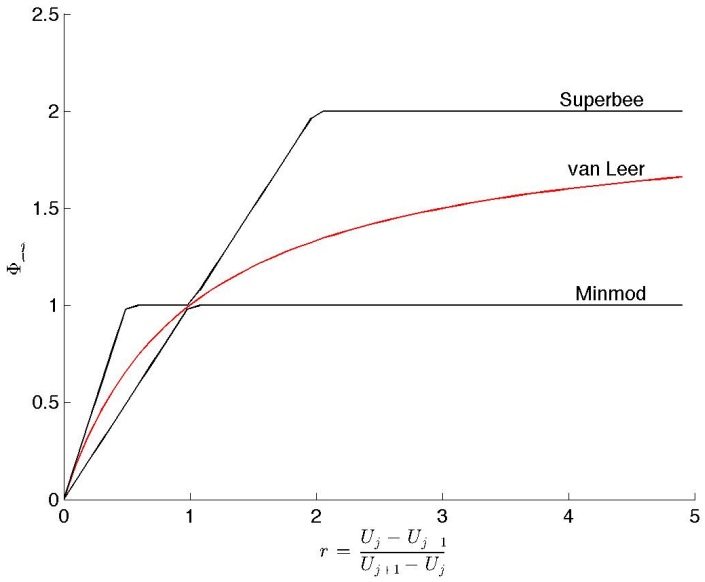

Figure 3 – Demonstrations of a linear interpolation scheme using a van Leer slope limiter; the black circles are the mean value in each of the cells, which are the regions between the vertical gray lines. The starred data points at each of the cell boundaries are the interpolated values of *c*. Note how at the extremum and discontinuities, the interpolation scheme becomes piecewise constant, thereby maximizing the diffusivity.

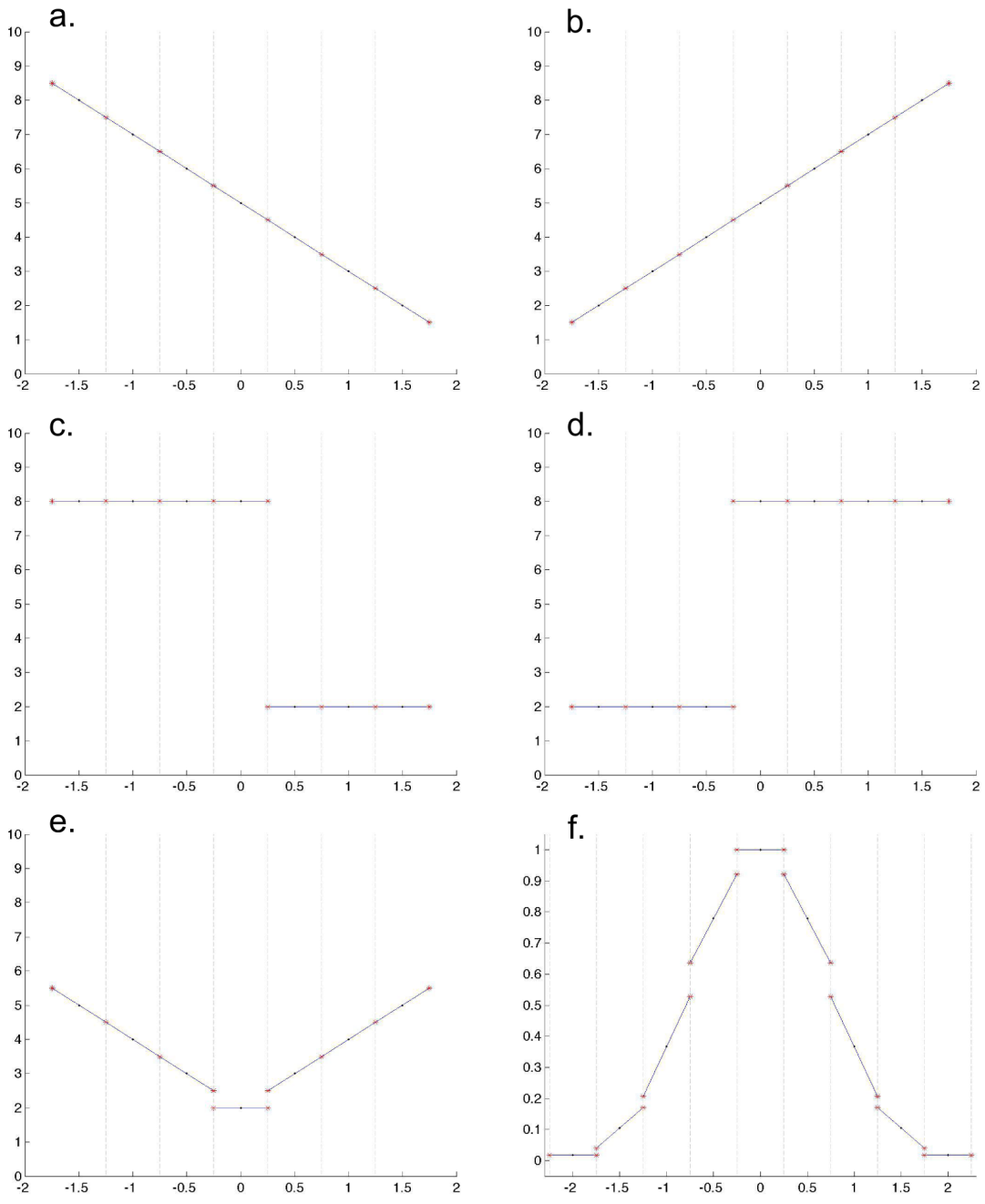

Figure 4 – Demonstration of the rise in the numerical dissipation at discontinuities when a linear interpolation scheme with slope limiters is used along with the Lax-Friederichs method. The final figure is normalized magnitude of the diffusive flux.

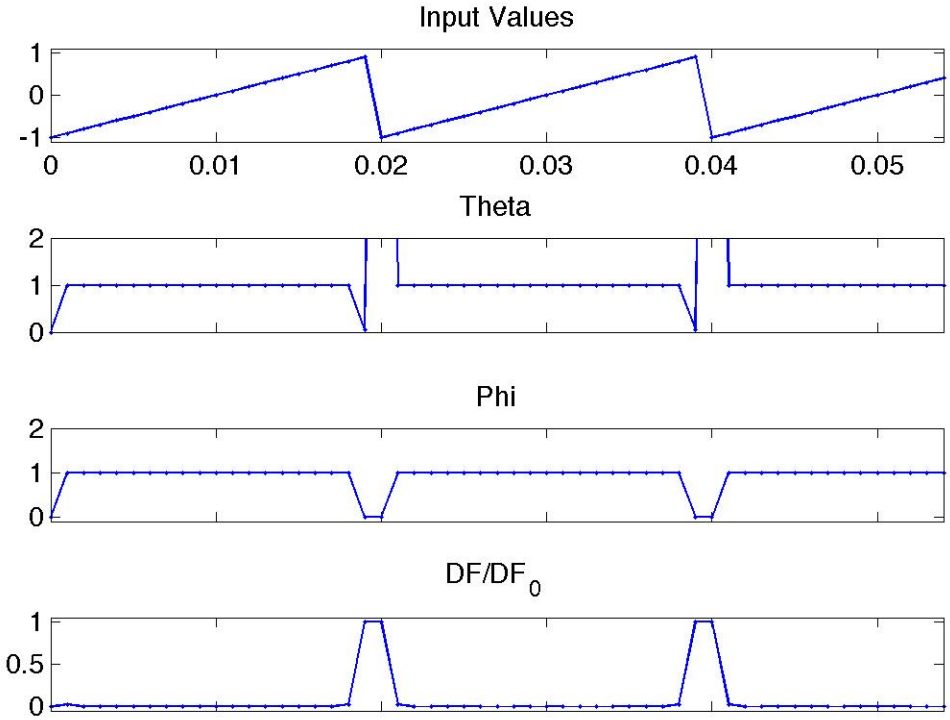

Figure 5 – The numerical approximation to the same problem studied in Figure 1 calculated using the local Lax-Friederichs scheme with slope limiters. The difference in accuracy seen by incorporating the slope limiters is very apparent.

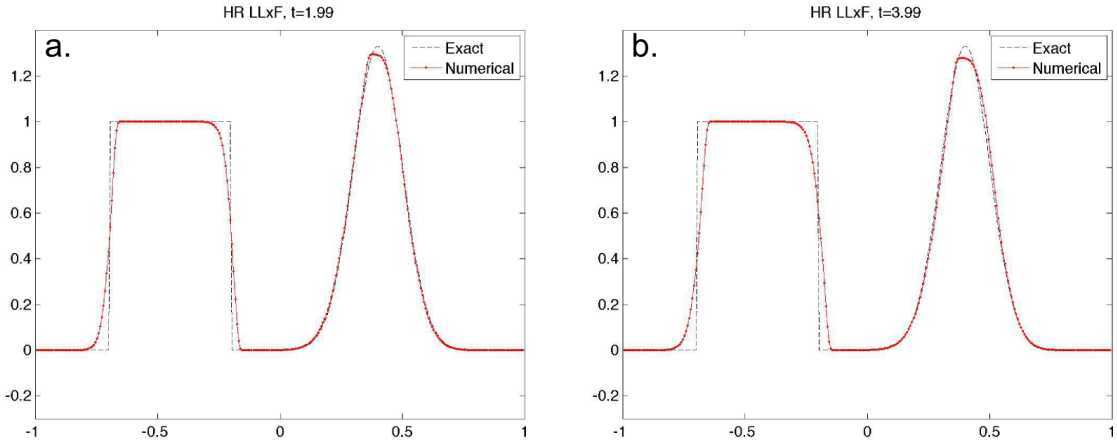

Figure 6 – Linear advection-diffusion simulation results using an explicit centered finite difference method. Figures a-c are from simulations in 1-D and contain the analytical solution. Figures d-i are from the 2-D simulations. The error analysis in the final figure confirms that the method is 2nd order accurate in both 1- and 2-D as expected.

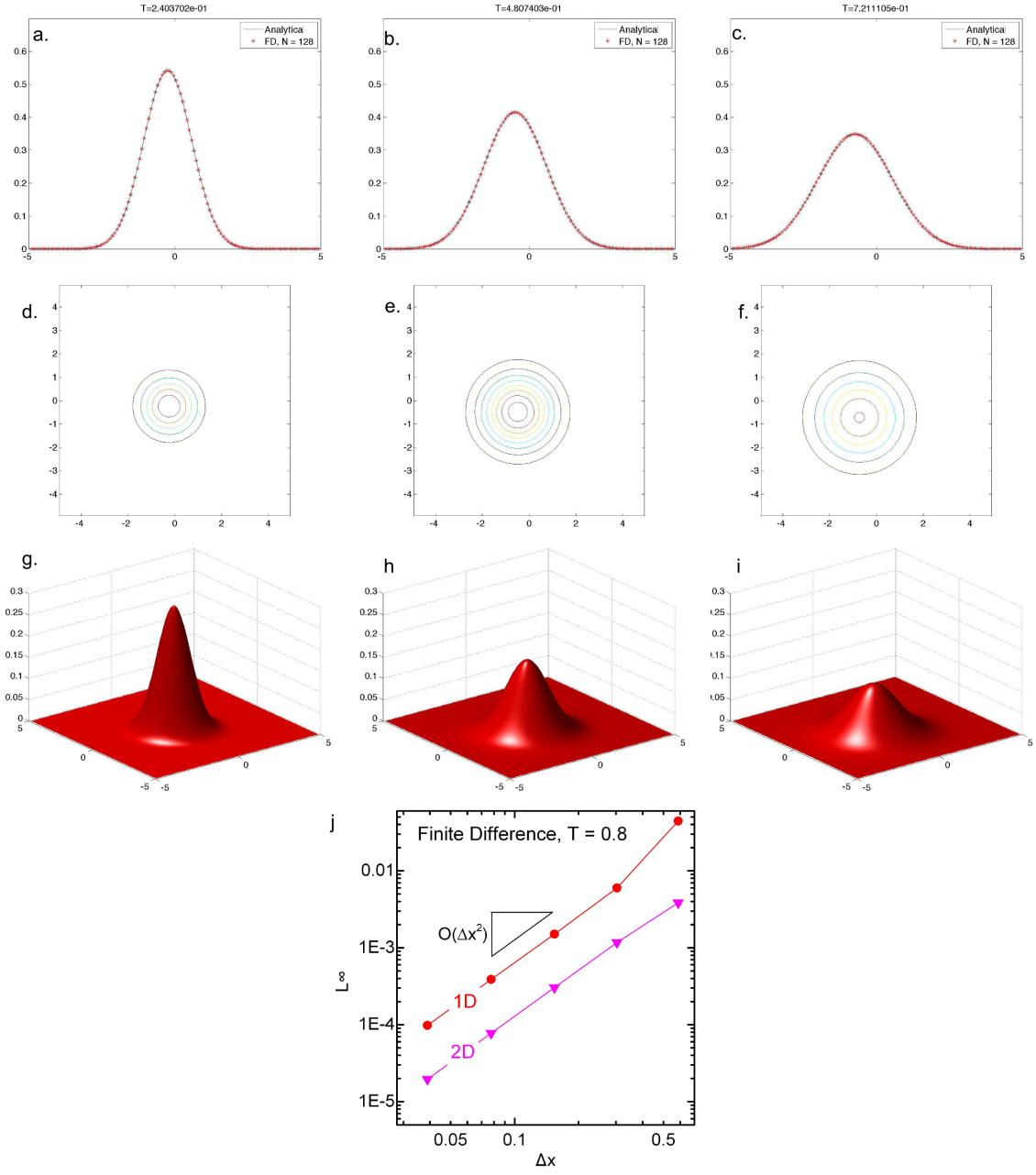

Figure 7 – Linear advection diffusion simulation results using the 2nd order accurate finite volume method. Figures a-c are from simulations in 1-D and contain the analytical solutions. Figures d-i are from simulations in 2-D. The error analysis in the final figure confirms that in 1-D the method is 2nd order accurate and that it is roughly 2nd order accurate in 2-D as well.

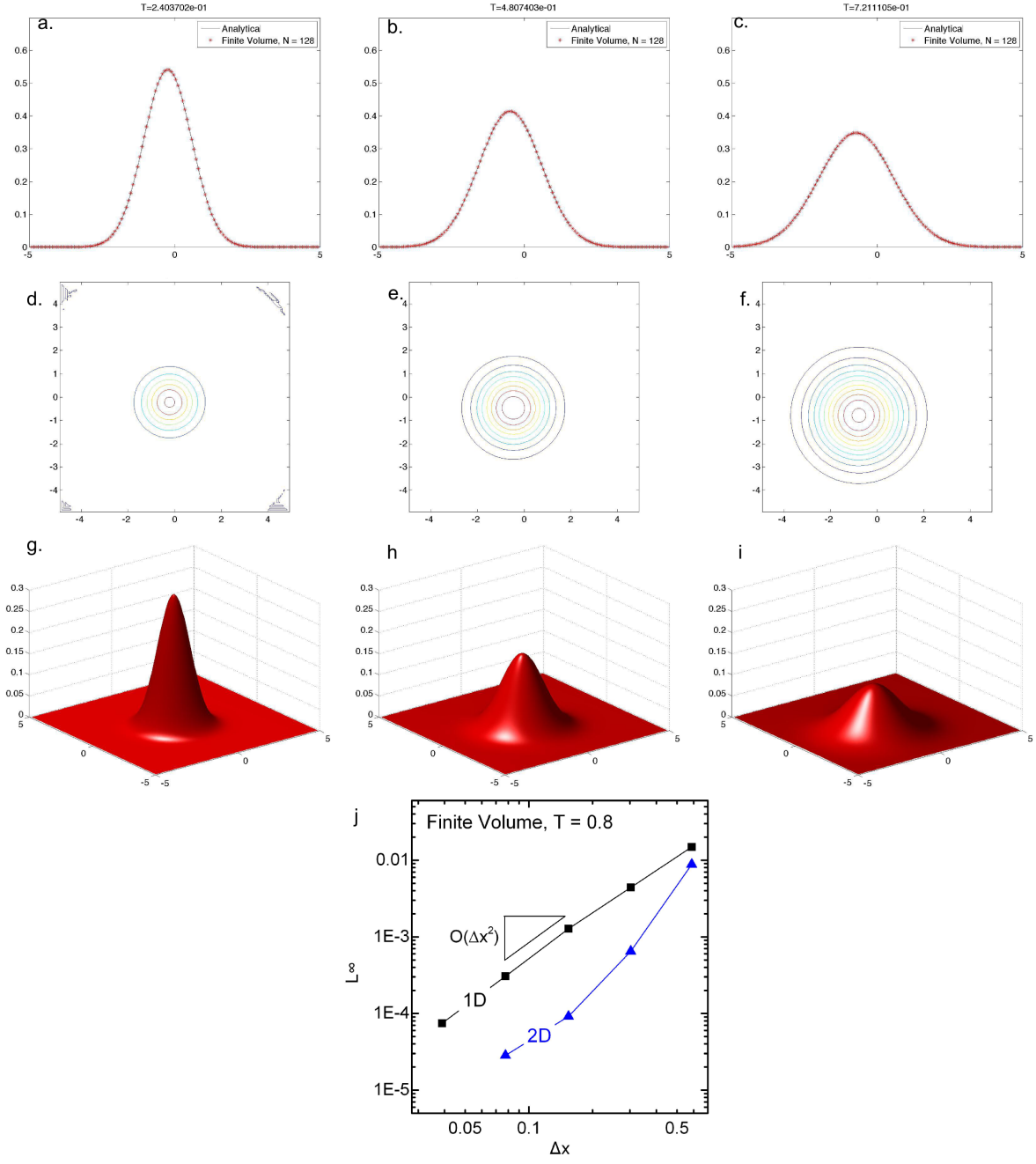

Table 1 – L^{∞} and L^{1} error table for the linear advection-diffusion problem.

	1-D				2-D			
N	**L^{∞}**	**L^{∞} Order**	**L^{1}**	**L^{1} Order**	**L^{∞}**	**L^{∞} Order**	**L^{1}**	**L^{1} Order**
16	1.47E-02	-	4.80E-03	-	8.80E-03		7.21E-04	
32	4.25E-03	1.79	1.17E-03	2.04	6.49E-04	3.76	1.52E-04	2.24
64	1.14E-03	1.90	3.12E-04	1.91	9.15E-05	2.83	4.23E-05	1.85
128	2.89E-04	1.97	8.26E-05	1.92	2.83E-05	1.69	1.20E-05	1.82
256	7.22E-05	2.00	2.13E-05	1.96	7.73E-06	1.87	3.21E-06	1.90
512	1.80E-05	2.00	5.39E-06	1.98				

Figure 8 – Non-linear flow of an initial slug of liquid through a porous medium under its own weight. Each row contains images from simulations using the exponent *m* and the time indicated. The cross-sections through the y-z plane in the final figure compare results from the three sets of simulations. The approximations are everywhere greater than or equal to 0 and the sharp, oscillation free interface between the "wet" and "dry" regions is encouraging.

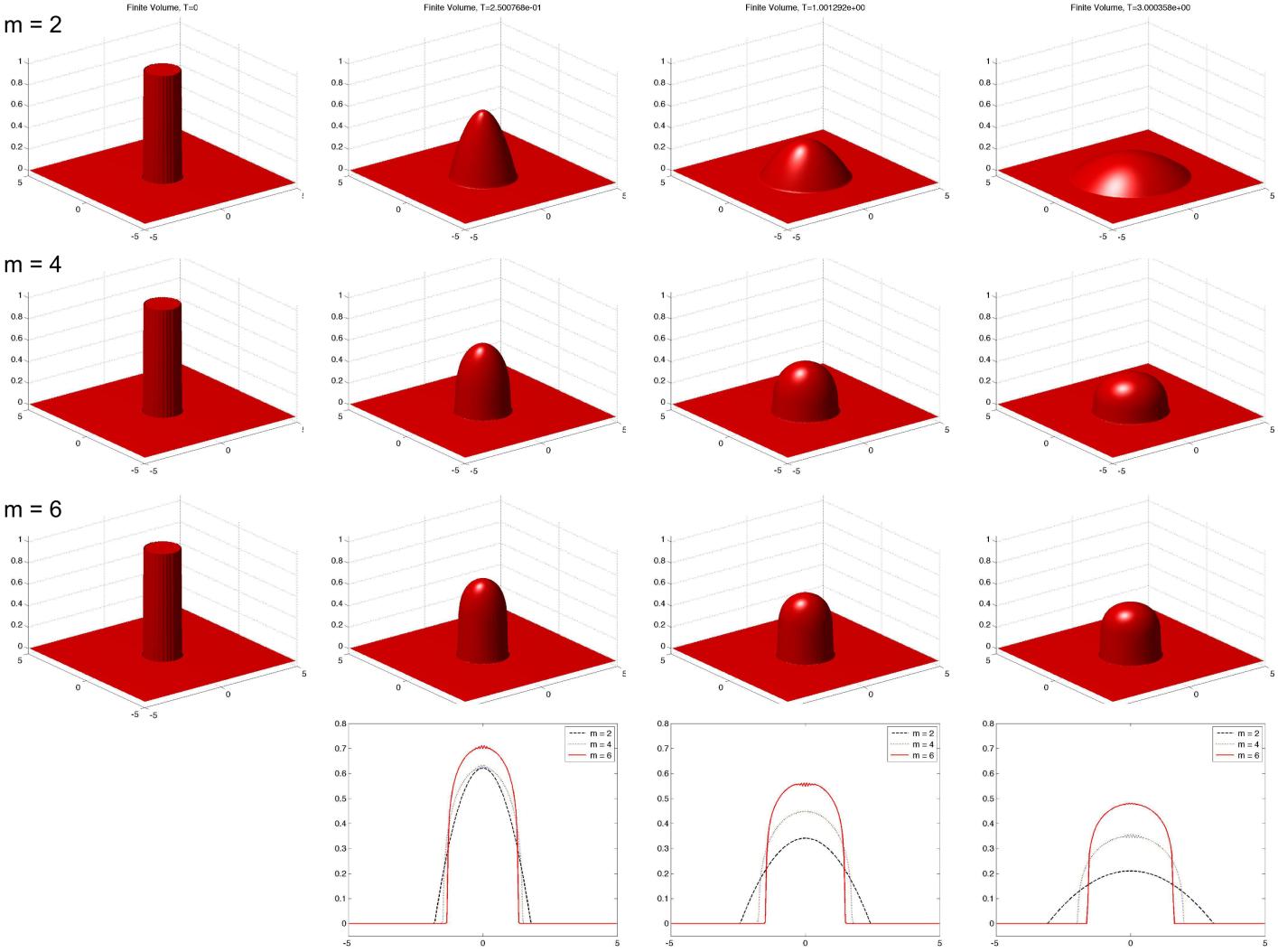

Figure 9 – Log of the radius of the liquid slug in the m = 2 simulations plotted against the log of the simulation time. A linear fit to this data has a slope of 0.22, which is slightly lower than expected (0.25) due to the starting condition.

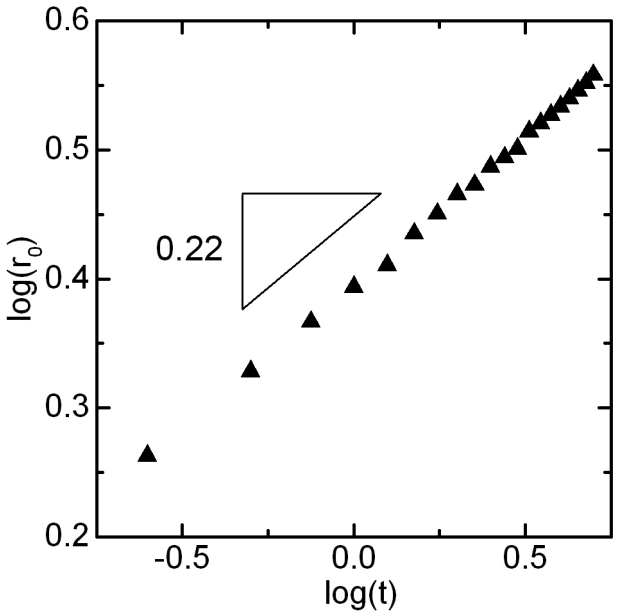

Figure 10 – Demonstration of the generalizability of this method by using it to solve a more complex advection-diffusion problem. Results match well with those presented in Ref [1].