Burgers numerical solution by Finite differences

James Woodfield

November 28, 2018

Abstract

We are solving burgers equation numerically. We will approach the problem with some naive ideas about how to resolve this problem and try and derive a method that is robust. To avoid trivial numerical schemes it is very important to note the following point; some numerical schemes add artificial diffusion. This is equivalent to changing the value of actual diffusivity ν to make the scheme more stable, making the inviscid limit case study convoluted.

Contents

1 Introduction

There are things to note about the burgers equation that make it interesting to study numerically, and there are different phenomena that occur in conjunction making the study of the problem confusing. Some properties actually arise from the numerics and not the pde, and in this project, we aim to study without the use of artificial diffusion.

1.1 Artificial diffusivity

1.1.1 Upwinding methodology

Methods noted as unwinding methods, actually introduce artificial diffusion into the problem. To demonstrate this consider the effect forwards in space has on a convective term in the convection convection equation (3). Consider the rearrangement below

$$u\delta_x^{\text{forwards}}\phi = u\frac{\phi_{j+1}-\phi_j}{\Delta x} = u\big(\frac{\phi_{j+1}-\phi_{j-1}}{2\Delta x} + \frac{2\phi_{j-1}-2\phi_j+\phi_{j+1}-\phi_{j-1}}{2\Delta x}\big) = u\delta_{2x}\phi + \frac{u\Delta x}{2}\delta_{xx}\phi$$

we have the centred in space, but with additional artificial diffusion. It is equivalent to solving the full convection diffusion with $-\nu \to -\nu + \frac{u\Delta x}{2}$ by the centred in space methods. This makes the study of the equation convoluted. It is for this reason we will avoid this method. Downwinding schemes have the same effect, and we avoid schemes like backwards in space.

1.1.2 Lax friedrichs

We can rewrite Lax friedrichs method, on the time derivative as

$$\delta_t^{\text{lax-freidrichs}}\phi = \frac{\phi_j^{n+1} - \frac{1}{2}(\phi_{j+1}^n + \phi_{j-1}^n)}{\Delta t} = \frac{\phi_j^{n+1}-\phi_j^n}{\Delta t} + \frac{-\phi_{j+1}^n + 2\phi_j^n - \phi_{j+1}^n}{2\Delta t} = \delta_t^{\text{forward}}\phi - \frac{\Delta x^2}{2\Delta t}\delta_{xx}(\phi)$$

it introduces artificial diffusion in a similar way, it would simpily change $\nu \to \nu + \frac{\Delta x^2}{2\Delta t}$ in the convection diffusion equation.

1.1.3 Other methodology

There are a wide variety of numerical methods that employ the notion of artificial diffusivity much more subtly with greater effect. Flux limiters, Godunov method and a variety of non-linear methods, have great properties and only introduce artificial diffusivity where needed. For example the flux limiter methods introduce diffusivity near the shocks where the gradient is large. Numerical code is already online for the implementation of such schemes, with experiments demonstrating the convergence properties. There are also other numerical methods when notions of introducing artificial diffusivity become hard to identify, particularly when working with the burgers equation.

1.2 Aims

We will try and work simpler numerical schemes with no artificial diffusion, and try and resolve shocks. We investigate methods to increase the timestep criteria. We also Investigate the different formulations of burgers equation.

2 Continuous Viscous Burgers equation

2.1 Equations

We will consider the Viscous burgers equation in one spatial dimension. We will study two forms of the equation, the non-conservative form

$$\frac{\partial \phi}{\partial t} + \phi \frac{\partial \phi}{\partial x} - \nu \frac{\partial^2 \phi}{\partial^2 x} = 0, \tag{1}$$

and the conservative form

$$\frac{\partial \phi}{\partial t} + \frac{\partial (\phi^2/2)}{\partial x} - \nu \frac{\partial^2 \phi}{\partial^2 x} = 0. \tag{2}$$

Where t is time, x is the spatial dimension, ϕ is the velocity in this direction, ν is viscosity. We bear both forms in mind as they result in different numerical schemes.

We will also work with another equation, called the convection diffusion equation

$$\frac{\partial \phi}{\partial t} + u \frac{\partial \phi}{\partial x} - \nu \frac{\partial^2 \phi}{\partial^2 x} = 0, \tag{3}$$

we do this as the Von-Neumann stability analysis of this equation helps with the study of the other two equations.

2.2 Boundary conditions

We will work with periodic boundary conditions on [0,1].

2.3 Initial conditions

$$\phi^0_{IC}(x) = \begin{cases} 1 & 0.1 \leq x \leq 0.3 \\ 0 & \text{else} \end{cases}$$

3 Discretization

3.1 Domain

For all numerics, we divide our domain into $nx = (\Delta x)^{-1} + 1$ equally spaced points on the interval $x \in [0,1]$ and have $nt = (\Delta t)^{-1} + 1$ equally spaced points in the interval $t \in [0,1]$.

3.2 Space component of the pde

We note to avoid smoothing the solutions by adding artificial diffusivity(or a higher order notion somewhat equivalent), we cant work with anything other than centred difference methods in space,

$$\frac{\partial U}{\partial x} \approx \delta_{2x} U := \frac{(U_{j+1} - U_{j-1})}{2\Delta x}, \quad \frac{\partial^2 U}{\partial x^2} \approx \delta_{xx} U := \frac{(U_{j+1} - 2U_j + U_{j-1})}{2\Delta x}.$$

these methods are 2nd order in space. So our scheme will be (It isn't demonstrated numerically because shocks will introduce Gibbs phenomena).

3.3 Time component of the pde

We Initially have no idea around stability so use the theta method in time, so we can switch to implicit methods if our scheme is unstable.

3.4 Boundary conditions

The effect of the periodic boundary conditions will become apparent in the matrix created by the discretised pde.

4 Stability: the Convection-Diffusion equation

Both forms of the burgers equations, are non-linear, this presents some difficulty when trying to assess whether the numerical schemes are well posed, in the sense that errors do not blow up. We instead study the convection diffusion equation (3) as it closely resembles the burgers equation, and is linear and suitable for Von-Neumann analysis.

4.1 Discretization

In this chapter we will defined the following constants $C = \frac{u\Delta t}{2\Delta x}$, $D = \frac{\nu \Delta t}{\Delta x^2}$. and discretise the convection diffusion equation to give the scheme

$$\phi_j^{n+1} + C\theta(\phi_{j+1}^{n+1} - \phi_{j-1}^{n+1}) - D\theta(\phi_{j+1}^{n+1} - 2\phi_j^{n+1} + \phi_{j-1}^{n+1}) \tag{4}$$
$$= \phi_j^n - C(1-\theta)(\phi_{j+1}^n - \phi_{j-1}^n) + D(1-\theta)(\phi_{j+1}^n - 2\phi_j^n + \phi_{j-1}^n). \tag{5}$$

4.2 Convection Diffusion: summary of Von-Neumann

Letting $\phi_j^n = A^n e^{ik(j)\Delta x}$ and $s = \sin(\frac{k\Delta x}{2})$ $c = \cos(\frac{k\Delta x}{2})$ our scheme gives

$$|A|^2 = \frac{((1-4D(1-\theta)s^2)^2) + (4C(1-\theta)sc)^2)}{\Big((1+4D\theta s^2)^2 + (4C\theta sc)^2\Big)}$$

for stability we require $|A|^2 \leq 1$ which holds iff $2(D^2s^2 + C^2c^2)(1-2\theta) \leq D$. If $\theta \geq 0.5$, the method is stable. If $\theta < 0.5$, the method is conditionally stable, in terms of the time step the condition for stability is

$$\Delta t \leq \min\Big\{\frac{\Delta x^2}{2\nu(1-2\theta)}, \frac{2\nu}{u^2(1-2\theta)}\Big\}. \tag{6}$$

In terms of the number of spatial nx and temporal nt points we have the condition $nt \geq \max\{2\nu(1-\theta)nx^2, \frac{u^2(1-\theta)}{2\nu}\}$.

5 Picards Method

We switch back to the equation of interest, burgers equation (1), (2), the equation is non linear. To solve it numerically we must turn this equation linear. Picards method as described in the literature on the numerical solution to the navier stokes equation approximates $\underline{u}^{n+1} \cdot \nabla \underline{u}^{n+1} \approx \underline{u}^n \cdot \nabla \underline{u}^{n+1}$ [1]. We replicate this by in the convective form and consider $\phi^{n+1}\frac{\partial \phi^{n+1}}{\partial x} \approx \phi^n \frac{\partial \phi^{n+1}}{\partial x}$. We now redefine $C = \frac{\Delta t}{2\Delta x}$, and write the non conservative θ method.

5.1 Non Conservative Picard's method

$$\phi_j^{n+1} + C\theta\phi_j^n(\phi_{j+1}^{n+1} - \phi_{j-1}^{n+1}) - D\theta(\phi_{j+1}^{n+1} - 2\phi_j^{n+1} + \phi_{j-1}^{n+1}) \tag{7}$$
$$= \phi_j^n - C(1-\theta)\phi_j^n(\phi_{j+1}^n - \phi_{j-1}^n) + D(1-\theta)(\phi_{j+1}^n - 2\phi_j^n + \phi_{j-1}^n). \tag{8}$$

the blue term here is to show the effect of the linearisation. ϕ_j^n is the value at position $x = 0 + j\Delta x$ at time $t = 0 + n\Delta t$.

Now we consider putting this into a matrix form, $A\phi^{n+1} = \beta^n$, noting A is different at every timestep. We infact put most of the schemes implemented in this project into a matrix form, so remember the banded structure of the matrix, that is always produced.

$$A = \begin{bmatrix} a_0 & b_0 & 0 & \dots & 0 & c_0 \\ c_1 & \ddots & \ddots & 0 & \dots & 0 \\ 0 & \ddots & & & & \\ \vdots & & & & \ddots & 0 \\ 0 & & & \ddots & \ddots & b_{nx-2} \\ b_{nx-1} & & & 0 & c_{nx-1} & a_{nx-1} \end{bmatrix}$$

Where:

$$a_j = 1 + 2\theta D, \quad b_j = C\theta\phi_j^n - D\theta, \quad c_j = -C\theta\phi_j^n - D\theta$$

$$\beta_j^n = \phi_j^n - C(1-\theta)\phi_j^n(\phi_{j+1}^n - \phi_{j-1}^n) + D(1-\theta)(\phi_{j+1}^n - 2\phi_j^n + \phi_{j-1}^n).$$

5.2 Conservative Picard's method

Here we linearise the conservative form of the pde by picards approximation before the derivative is taken through.

$$\phi_j^{n+1} + C\theta\frac{1}{2}[\phi_j^n(\phi_{j+1}^{n+1} - \phi_{j-1}^{n+1}) + \phi_j^{n+1}(\phi_{j+1}^n - \phi_{j-1}^n)] - D\theta(\phi_{j+1}^{n+1} - 2\phi_j^{n+1} + \phi_{j-1}^{n+1}) \tag{9}$$

$$= \phi_j^n - C(1-\theta)\frac{1}{2}((\phi_{j+1}^n)^2 - (\phi_{j-1}^n)^2) + D(1-\theta)(\phi_{j+1}^n - 2\phi_j^n + \phi_{j-1}^n). \tag{10}$$

blue term here is a term that has been highlighted to show the effect of the linearisation. Lets consider How this is put in the matrix form $A\phi^{n+1} = \beta^n$

$$a_j = 1 + 2D\theta + \frac{C\theta}{2}(\phi_{j+1}^n - \phi_{j-1}^n), \quad b_j = \frac{\theta C\phi_{j+1}^n}{2} - D\theta, \quad c_j = \frac{-\theta C\phi_{j-1}^n}{2} - D\theta,$$
$$\beta^n = \phi_j^n - C(1-\theta)\frac{1}{2}((\phi_{j+1}^n)^2 - (\phi_{j-1}^n)^2) + D(1-\theta)(\phi_{j+1}^n - 2\phi_j^n + \phi_{j-1}^n),$$

5.2.1 Numerical tests

Throughout this section we will consider the Von-Neumann analysis of the convection diffusion equation(3), and how it relates to burgers equation(1)(2). We will work with picards method $\theta = 0$ on the conservative form, this is known as forwards in time centred in space scheme, or ftcs for short.

Why this method: The picards approximation isn't used at all for $\theta = 0$. The matrix A has become the identity, and I code this up as a special case and the code runs very fast. There are two things to try and remember: the condition on stability (6), and whether the numerical scheme captures the shock, and this normally depends upon the spatial resolution.

5.2.2 Ideal parameters: figure 1a)

Choosing $\mu = 10^{-3}$ and $\Delta x = 10^{-3}$ we get the spatial discretisation very small to capture the shock, and not so coincidently this also gives the generous bound on the timestep $\Delta t < \frac{1}{2000}$. So we choose 1001 spatial points, and 2001 time points. To have a look see figure 1a). It has worked remarkably well, the shock is captured by the large number of spatial points, and the time-step isn't cruelly big.

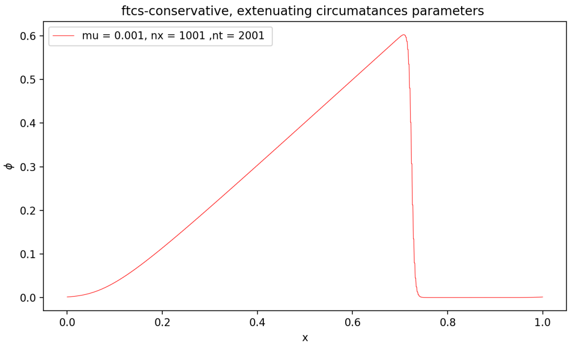
5.2.3 Not so Ideal parameters: figure 1b),1c)

In the last example we picked helpful values of ν and Δx based on the Von-Neumann analysis of a similar equation called the convection diffusion equation. We study two numerical examples below.

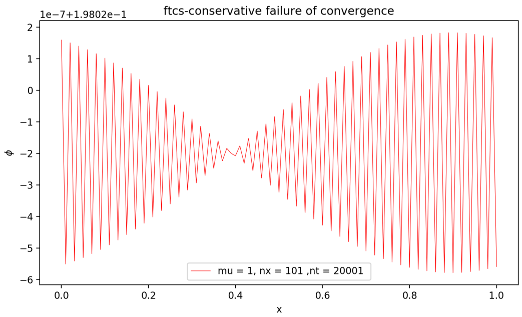
Figure 1b, we try and solve the same burgers equation, by the same ftcs scheme, but with parameters $\nu = 1$, $nx = 101$, $nx = 20001$. We get instability. Conclusion: this case shows the effect the condition $\Delta t \leq \frac{\Delta x^2}{2\nu}$ can have, even with moderate Δx.

Case 1c) we consider the other case, when $\nu = 0.0001$ $\Delta x = 0.01$ $\Delta t = \frac{1}{20001}$ Here we satisfy the first condition that $\Delta t \leq \frac{\Delta x^2}{2\nu}$, but only just the second condition $\Delta t \leq 2\nu$. The numerical result is found in figure 1c). Conclusion: we have two things to take away from this example. We firstly note how the value of μ alone can put some heavy restrictions on the timestep, independent of spatial discretisation. The second thing to take away is the fact that even staying within these time stepping conditions isn't good enough. We stayed within stability conditions and our numerics were unreliable, because the spatial discretisation is too coarse as compared to the shock interface.

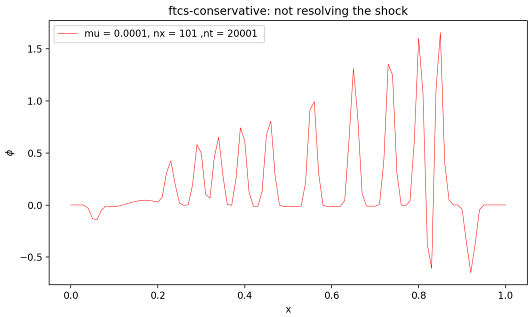
Overall we learned $\Delta x \approx O(\nu)$ for shock capture, so we dont get figure 1c), and also so the time stepping conditions to be optimal. This was how I magically picked numbers to create 1a).



(a)



(b)

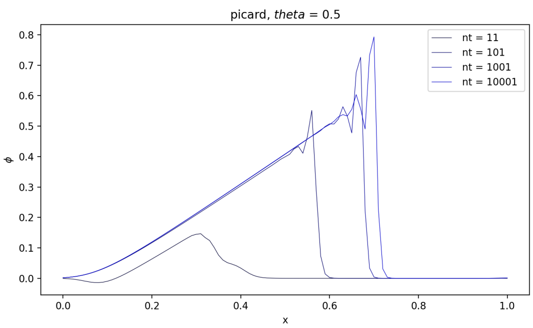


(c)

Figure 1: All three are ftcs on conservative the burgers equation i.e. conservative picard with $\theta = 0$. a) Very succesful capture of shock with $\nu = 0.001$, $nx = 1001$, $nt = 2001$. b) Failure of convergence with the values $\nu = 1$, $nx = 101$, $nt = 20001$ c) Incorrect shock capturing for the values $\nu = 0.0001$, $nx = 101$, $nt = 20001$.

5.2.4 Picards method for $\theta = 0.5$

We consider Picards $\theta = 0.5$ method on the non conservative form of burgers equation. We use $(\Delta x)^{-1} = nx = 101$, $\nu = 0.001$ and plot the solutions obtained from a different number of timesteps: $nt = (\Delta t)^{-1} = 11, 101, 1001, 10001$. To see the results see figure 2a).



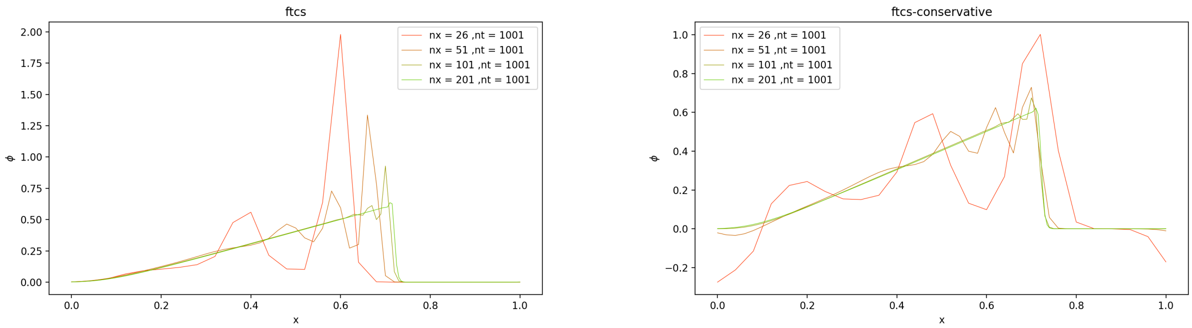
(a)

Figure 2: The figure 2a) gives shows the picards approximation as timestep increases.

Conclusions from the numerics. The scheme remains stable even for unruly time steps, but not accurate. Explanation, for $\theta = 0.5$ we have unconditional stability in the convection diffusion equation, the equation we work with is very similar. The scheme is inaccurate because we linearised the burgers equation by approximating $\phi^{n+1} \approx \phi^n + O(\Delta t)$, key word is approximating. This method would be equivalent to the crank-nicholson method, however we lose second order accuracy in time because picards aproximation is first order in time. Other Conclusions to be drawn from figure 2a). The spatial discretisation is too small to completely resolve the shock exactly.

5.2.5 Numerical results: Conservative vs Non-Conservative shock resolving

We now investigate how the spatial resolution deals with the shock, we work with $\theta = 0$ for conservative and non conservative forms of the burgers equation. We work with $nt = 1001$, $\nu = 0.001$ and have $nx = 26, 51, 101, 201$.



(a) (b)

Figure 3: The figure 3a) gives the ftcs scheme as spatial discretisation increases, the figure 3b) gives the conservative ftcs scheme as spatial discretisation increases.

Figure 3a) and figure 3b) show how doubling the spatial resolution improves shock capturing. This is a useful result, provides motivation for mesh adaptivity. Another conclusion is that the conservative discretisation, captures the shock profile location very well at every spatial resolution.

Error analysis of these equations is misleading, different norms would tell you very different things. The graphs are more informative. The global error of the conservative form is higher and the local error near the shock is lower than the non-conservative form. Accurate shock capturing introduces error globally, and this is a fundamental property of the conservative form.

6 Newton method

6.1 Introduction

This problem is non-linear in ϕ^{n+1} for $\theta \neq 0$, we will be using newtons method to linearise. We assume we already have ϕ^n. write in general form $F(w) = 0$ and construct a sequence $\{w^k\}$ that converges quadratically to w.

$$F'(w^k)(\delta w) = -F(w^k), \quad \text{where} \quad \delta w = w^{k+1} - w^k, \quad w^0 = \phi^n.$$

We will get a matrix form to solve at every timestep, so write $F(w^k) = A_k$ and $-F(w^k) = \beta_k$.

6.1.1 Pseudo Code

Set initial condition $\phi_j^0 = IC$
While $n \leq nt$:
    n = n+1
    Set $w_j = \phi_j^n$, $\forall j$
    While error > tolerance :
        Compute A_k and β_k
        solve $A_k \delta w = \beta_k$ for δw
        update $w = \delta w + w$
        Set error $= ||\delta w||_{L^2}$
    Update $\phi_j^{n+1} = w$ (we have just found the next timestep, within some tolerance)
We have used the almost ridiculous tolerance of tol $= 10^{-13}$ in the newton method, to enforce accuracy.

6.2 Non conservative form

$$\delta w_j + \theta C \delta w_j (w_{j+1}^k - w_{j-1}^k) + C\theta w_j^k (\delta w_{j+1} - \delta w_{j-1}) - D\theta(\delta w_{j+1} - 2\delta w_j + \delta w_{j-1})$$
$$= \phi_j^n - C(1-\theta)\phi_j^n(\phi_{j+1}^n - \phi_{j-1}^n) + D(1-\theta)(\phi_{j+1}^n - 2\phi_j^n + \phi_{j-1}^n) - w_j^k$$
$$- C\theta w_j^k(w_{j+1}^k - w_{j-1}^k) + D\theta(w_{j+1}^k - 2w_j^k + w_{j-1}^k)$$

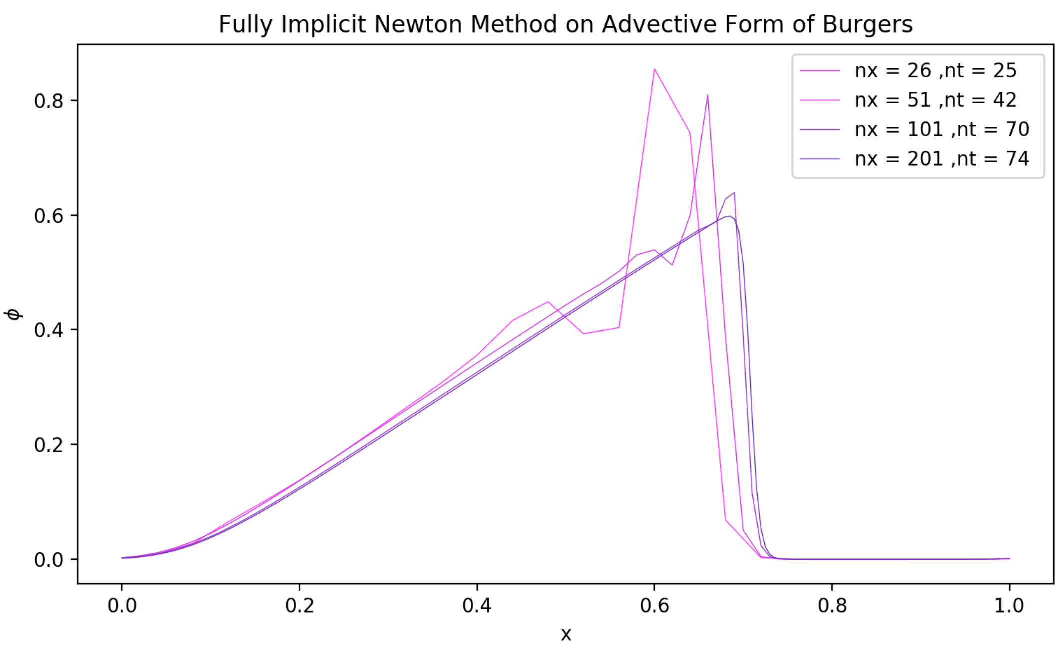
We now consider writing it in the form A_k, β_j similar to what we did for picards method.

$$a_j = 1 + C\theta(w_{j+1}^k - w_{j-1}^k) + 2D\theta, \quad b_j = C\theta w_j^k - D\theta, \quad c_j = -C\theta w_j^k - D\theta,$$
$$\beta_j^k = \phi_j^n - C(1-\theta)\phi_j^n(\phi_{j+1}^n - \phi_{j-1}^n) + D(1-\theta)(\phi_{j+1}^n - 2\phi_j^n + \phi_{j-1}^n) - w_j^k$$
$$- C\theta w_j^k(w_{j+1}^k - w_{j-1}^k) + \theta D(w_{j+1}^k - 2w_j^k + w_{j-1}^k)$$

In this method we use modular nx arithmetic to implement periodic boundary conditions.

6.3 Numerics

Figure 4a) demonstrates how this method captures the shocks as spatial resolution varies. We use $\theta = 1$.



(a)

Figure 4: We have $\theta = 1$, implicit newton method, for increasing spatial resolution on the non conservative equation

6.3.1 Numerical results: shock capturing, figure 4

We run a similar experiment to before. We compare figure 4 to the figure 3a) to compare the non conservative newton method to the non conservative ftcs scheme. How does newton compare to ftcs? The newton method has smaller errors globally and the shock is captured more accurately as well as the time-step condition is drastically reduced. It is interesting to numerically see that the newton methodology has helped with some of the Gibbs like phenomena. The drawback is the implementation is much more computationally expensive and difficult.

6.4 Conservative form

The θ conservative method is first obtained by a finite volume consideration.

$$\frac{\phi_j^{n+1}-\phi_j^n}{\Delta t}+\theta(\frac{(\phi^2)_{j+1}^{n+1}-(\phi^2)_{j-1}^{n+1}}{4\Delta x}-\nu\frac{\phi_{j+1}^{n+1}-2\phi_j^{n+1}+\phi_{j-1}^{n+1}}{\Delta x^2})=$$
$$-(1-\theta)(\frac{(\phi^2)_{j+1}^{n}-(\phi^2)_{j-1}^{n}}{4\Delta x}-\nu\frac{\phi_{j+1}^{n}-2\phi_j^{n}+\phi_{j-1}^{n}}{\Delta x^2})$$

Then after the newton linearisation we have the below form:

$$\begin{aligned}&\delta w_j+C\theta(w_{j+1}^k\delta w_{j+1}-w_{j-1}^k\delta w_{j-1})-D\theta(\delta w_{j+1}-2\delta w_j+\delta w_{j-1})\\&=\phi_j^n-C(1-\theta)\frac{(\phi^2)_{j+1}^n-(\phi^2)_{j-1}^n}{2}+D(1-\theta)(\phi_{j+1}^n-2\phi_j^n+\phi_{j-1}^n)\\&-w^k-C\theta\frac{(w^2)_{j+1}^k-(w^2)_{j-1}^k}{2}+D\theta(w_{j+1}^k-2w_j^k+\delta w_{j-1}^k)\end{aligned}$$

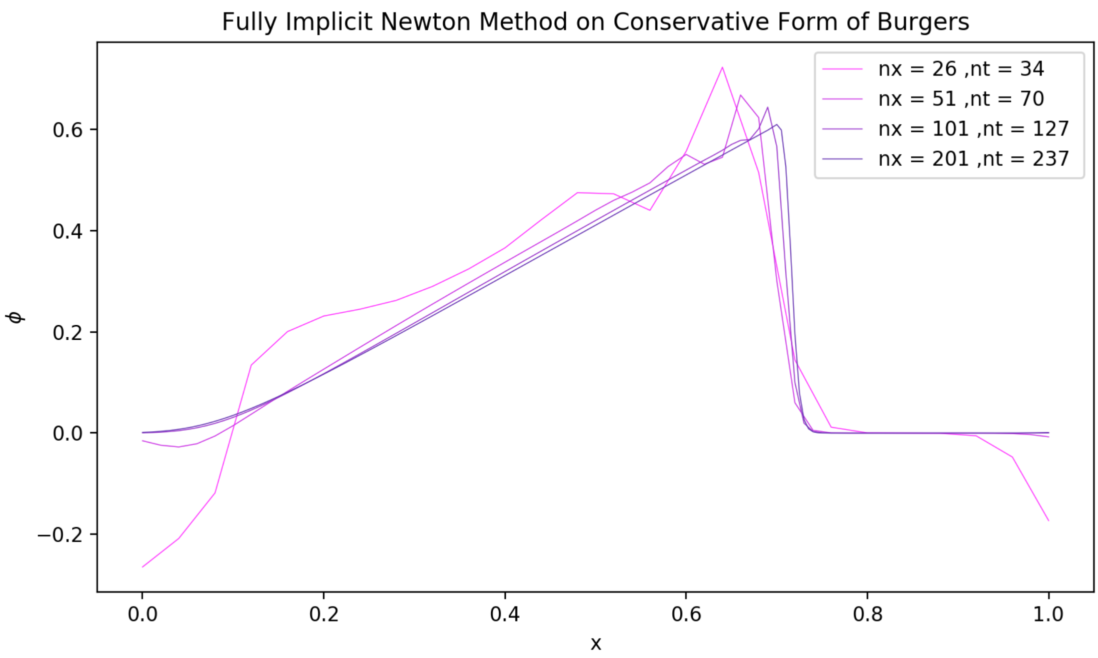
We now consider writing it in the matrix form A_k, β_j.

$$\begin{aligned}&a_j=1+2D\theta,\quad b_j=C\theta w_{j+1}^k-D\theta,\quad c_j=-C\theta w_{j-1}^k-D\theta,\\&\beta_j^k=\phi_j^n-C(1-\theta)\frac{(\phi^2)_{j+1}^n-(\phi^2)_{j-1}^n}{2}+D(1-\theta)(\phi_{j+1}^n-2\phi_j^n+\phi_{j-1}^n)\\&\quad-w_j^k-C\theta\frac{(w^2)_{j+1}^k-(w^2)_{j-1}^k}{2}+D\theta(w_{j+1}^k-2w_j^k+\delta w_{j-1}^k)\end{aligned}$$

In this method we use modular nx arithmetic to implement periodic boundary conditions.

6.5 Numerics

Figure 5a) demonstrates how this method captures the shocks as spatial resolution varies. Similarly to how we did for the ftcs-conservative scheme. We use $\theta = 1$.



(a)

Figure 5: We have $\theta = 1$, implicit newton method, for increasing spatial resolution on conservative burgers equation.

6.5.1 Shock capturing: figure 5

Compare the newly generated solution 5a) to figure 4a), and it is apparent the phenomena for the conservative form is carried through the newton methodology. The shock is calculated closer to the actual shock, and there are more global errors, and less local errors. For example just behind the shock consider the error of the figure.

7 Comparisons

7.1 Timestepping

The newton method isn't guaranteed to converge from the initial guess. Therefore we have a time step restriction independent of the one derived earlier. It is a well known theorem in mathematics that it' impossible to know whether a newton solver will converge given random initial conditions. However, In this section, we demonstrate increasing the spatial resolution effects the time stepping criterion. And more subtle point is demonstrated numerically, the conservative form requires smaller time steps to converge.

We have computed the previous two examples, non conservative and conservative newtons method on the burgers equation as spatial resolution varies. Behind the scenes, I trialled various temporal resolutions for every choice of spatial discretisation, and compiled the smallest number of timesteps needed to get convergence out the methods implemented. I put these values in a table see (Table 1).

nx	nt (Non-Conservative)	nt (Conservative)
26	25	34
51	42	70
101	70	127
201	74	237

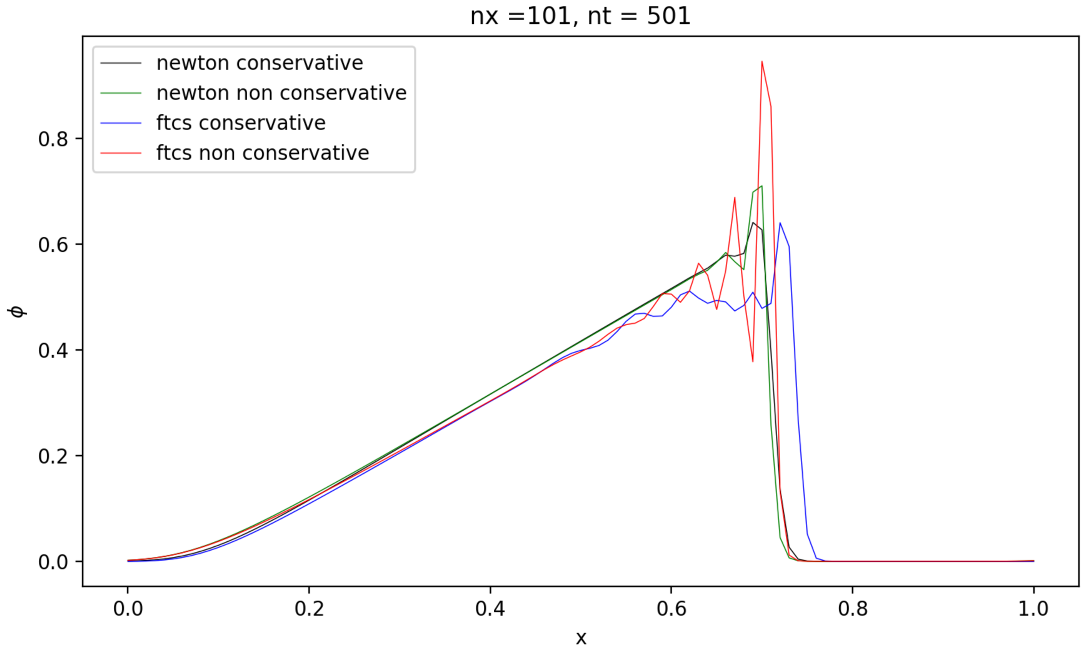
Table 1: The smallest values for nt in which a newton method converges

Observations from the table. The time steps are much greater than those for the explicit methods where in the best case 501 points are required for convergence. Increasing the number of space points increases the number of time points needed, considering that doubling the number of space points, doubles the required number of time points is very interesting and promising compared to the previous bound $\Delta t < O(\Delta x^2)$. Now we have the conservative form is more unstable than the non conservative form, this is because the form inherently tracks the shocks location better, even with the newton methodology. This essentially puts more constraints on the newton problem to converge, and so requires more time steps. And as it should, these extra constraint that you track the shock accurately result in a more accurate solution as observed when comparing figures 4 and 5.

8 comparison of all methods

References

[1] H.C. Elman, D.J. Silvester, A.J. Wathen, *Finite Elements and Fast Iterative Solvers: with Applications in Incompressible Fluid Dynamics*, second ed., Oxford University Press, 2014.



(a)

Figure 6: A comparison of all main methods at the limit of ftcs stability. Even with convergence, the newton methods do better. Here the conservative ftcs starts to show blow up.