12 The linear diffusion equation

The linear diffusion equation is

$$\frac{\partial \phi}{\partial t} = K \frac{\partial^2 \phi}{\partial x^2}$$

with K a constant. Assume the domain is periodic in x. If the initial condition is wavelike

$$\phi(x, 0) = e^{ikx} \quad \text{(real part understood)}$$

then the exact solution can be found. Let

$$\phi(x, t) = \Phi(t) e^{ikx}.$$

Then

$$\frac{d\Phi}{dt} = -k^2 K \Phi,$$

so

$$\Phi = \Phi(0) e^{-k^2 K t}; \quad \Phi(0) = 1$$

$$\phi(x, t) = e^{-k^2 K t} e^{ikx}.$$

The solution is a stationary wave decreasing in amplitude. The damping becomes faster as the wavelength decreases.

12.1 Stability of some example schemes—von Neumann stability analysis

Divide the periodic space domain into N equal intervals of size Δx so that $x_j = j\Delta x$, $j = 0,\ 1,\ 2, \ldots,\ N$. Choose a time step Δt and let $t^{(n)} = n\Delta t$, $n = 0,\ 1,\ 2,\ \ldots$. Write ϕ_j^n for the numerical approximation to $\phi(x_j, t^{(n)})$.

Consider first a centred difference scheme:

$$\frac{\phi_j^{(n+1)} - \phi_j^{(n-1)}}{2\Delta t} = K \left(\frac{\phi_{j+1}^{(n)} - 2\phi_j^{(n)} + \phi_{j-1}^{(n)}}{\Delta x^2} \right).$$

This scheme is second order in space and second order in time, meaning that if we substitute the true solution into the finite difference scheme then the dominant terms in the residual are a term proportional to Δx^2 and a term proportional to Δt^2.

Von Neumann stability analysis. On a *periodic domain* any function can be decomposed into Fourier components. If the evolution of the function is governed by a *linear equation* with *constant coefficients* then its behaviour can be determined by looking at the behaviour of each Fourier component. Similarly, the stability of a linear numerical method can be found be considering each Fourier component and checking whether it grows in time. More generally, for nonlinear equations or equations with non-constant coefficients or non-periodic boundary conditions, the stability of individual Fourier components gives a useful guide but does not provide a sufficient condition for the stability of the full problem.

Look for numerical solutions of the form

$$\phi_j^{(n)} = A^n e^{ikj\Delta x}$$

where A is a (possibly complex) constant—the *amplification factor*. If we find $|A| > 1$ then the numerical solution is growing with time.

Substitute into the numerical scheme to obtain

$$A^{n+1}e^{ikj\Delta x} - A^{n-1}e^{ikj\Delta x} = \frac{2K\Delta t}{\Delta x^2}\left(A^n e^{ik(j+1)\Delta x} - 2A^n e^{ikj\Delta x} + A^n e^{ik(j-1)\Delta x}\right).$$

Cancelling powers of A and $e^{ikj\Delta x}$ leaves a quadratic equation for A:

$$A^2 + \frac{2K\Delta t}{\Delta x^2}\left(2 - 2\cos k\Delta x\right)A - 1 = 0.$$

There are two roots for A, corresponding to the physical mode and the computational mode.

$$A = -\frac{2K\Delta t}{\Delta x^2}\left(1 - \cos k\Delta x\right) \pm \left[\left(\frac{2K\Delta t}{\Delta x^2}\right)^2 \left(1 - \cos k\Delta x\right)^2 + 1\right]^{1/2}.$$

Both roots A^+ and A^- are real and their product is $A^+A^- = -1$. A^+ lies between 0 and 1 and this corresponds to the physical mode ($A^+ \to 1$ as $\Delta x,\ \Delta t/\Delta x \to 0$). On the other hand, $A^- < -1$ and this corresponds to the computational mode. The computational mode is unconditionally unstable. Therefore, this centred difference scheme is a poor choice for the diffusion equation.

An alternative scheme uses a forward Euler rather than a centred time step:

$$\frac{\phi_j^{(n+1)} - \phi_j^{(n)}}{\Delta t} = K\left(\frac{\phi_{j+1}^{(n)} - 2\phi_j^{(n)} + \phi_{j-1}^{(n)}}{\Delta x^2}\right).$$

This is first-order accurate in time and second-order accurate in space.

Von Neumann analysis leads to the following expression for the amplification factor:

$$A = 1 - \frac{2K\Delta t}{\Delta x^2}\left(1 - \cos k\Delta x\right).$$

Now $0 \le (1 - \cos k\Delta x) \le 2$, so $|A| \le 1$ if $2K\Delta t/\Delta x^2 \le 1$. This scheme is conditionally stable. Note that, because of the quadratic dependence on Δx, a moderate increase in resolution might require a much smaller time step.

Another alternative scheme uses a backward Euler time step:

$$\frac{\phi_j^{(n+1)} - \phi_j^{(n)}}{\Delta t} = K\left(\frac{\phi_{j+1}^{(n+1)} - 2\phi_j^{(n+1)} + \phi_{j-1}^{(n+1)}}{\Delta x^2}\right).$$

Again, this is first-order accurate in time and second-order accurate in space.

Von Neumann analysis leads to

$$A = \left(1 + \frac{2K\Delta t}{\Delta x^2}\left(1 - \cos k\Delta x\right)\right)^{-1}.$$

For this scheme $|A| \le 1$ for any value of Δt, so it is unconditionally stable. However, the unknown $\phi^{(n+1)}$ appears on both sides of the equation defining the scheme, so it is non-trivial to solve for $\phi^{(n+1)}$. In this linear one-dimensional example $\phi^{(n+1)}$ can be found by solving a tridiagonal system of simultaneous equations, which can be done efficiently using Gaussian elimination.

13 The linear advection equation

Consider the equation

$$\frac{D\phi}{Dt} = \frac{\partial \phi}{\partial t} + u\frac{\partial \phi}{\partial x} = 0 \tag{20}$$

where u is a constant velocity, the domain is $0 \leq x \leq 1$ with periodic boundary conditions $\phi(0,t) = \phi(1,t)$, and the initial condition is given as $\phi(x,0) = F(x)$.

13.1 Exact solution and conservation properties

The solution of this problem is

$$\phi(x,t) = F(x-ut),$$

(where $x - ut$ is understood to be taken modulo 1 to account for the periodicity). The initial function is advected with speed u, and its shape is preserved.

Equation (20) has a number of invariants (actually an infinite number). For example, let

$$M = \int_0^1 \phi \, dx$$

Then

$$\frac{dM}{dt} = \int_0^1 \frac{\partial \phi}{\partial t} \, dx = -\int_0^1 u\frac{\partial \phi}{\partial x} \, dx = -u\left[\phi\right]_0^1 = 0$$

(because of the periodic boundary conditions). Thus, M is conserved. Similarly, multiplying (20) by ϕ gives

$$\frac{\partial \phi^2}{\partial t} + u\frac{\partial \phi^2}{\partial x} = 0. \tag{21}$$

Then a similar calculation to the above shows that $V = \int_0^1 \phi^2 \, dx$ is conserved.

It is sometimes desirable for a numerical scheme to have conservation properties analogous to those of the original equation.

13.2 The upstream scheme

Divide the periodic space domain into N equal intervals of size Δx so that $x_j = j\Delta x,\ j = 0,\ 1,\ 2,\ \ldots,\ N$. Choose a time step Δt and let $t^{(n)} = n\Delta t,\ n = 0,\ 1,\ 2,\ \ldots$. Let $\phi_j^{(n)}$ be the numerical approximation to $\phi(x_j, t^{(n)})$.

Consider a forward in time, backward in space (FTBS) approximation:

$$\frac{\phi_j^{(n+1)} - \phi_j^{(n)}}{\Delta t} + u\frac{\phi_j^{(n)} - \phi_{j-1}^{(n)}}{\Delta x} = 0$$

Von Neumann stability analysis gives an equation for the amplification factor. Let $\phi_j^{(n)} = A^n e^{ikj\Delta x}$. Then

$$A^{n+1}e^{ikj\Delta x} - A^n e^{ikj\Delta x} + \frac{u\Delta t}{\Delta x}\left(A^n e^{ikj\Delta x} - A^n e^{ik(j-1)\Delta x}\right) = 0.$$

Define $c = u\Delta t/\Delta x$; c is called the *Courant number*. Cancel powers of A and $e^{ikj\Delta x}$ to obtain

$$A - 1 + c\left(1 - e^{-ik\Delta x}\right) = 0.$$

The amplification factor is complex, as it must be for a propagating solution. The numerical solution will be stable if $|A| \leq 1$. $|A|^2$ is given by A times its complex conjugate. After some algebra

$$|A|^2 = 1 - 2c(1-c)(1 - \cos k\Delta x).$$

Now $(1 - \cos k\Delta x)$ is always greater than or equal to zero, so $|A|^2$ will be less than or equal to 1 provided $c(1-c) \geq 0$, i.e. provided $0 \leq c \leq 1$.

Note that the FTBS scheme is unstable for $u < 0$. But in that case we could switch to using a forward in time forward in space (FTFS) scheme. In the general case, for which u could take either sign, we must use FTBS when $u > 0$ and FTFS when $u < 0$, so that we always use information from the upstream side of the point whose new value we are trying to calculate. This combined FTBS/FTFS scheme is called the *upstream scheme*.

When the time step is small enough so that the upstream scheme is stable, $|A| \leq 1$. For the numerical scheme to reproduce the true solution we would need $|A| = 1$. In general the numerical solution is damped. Damping error, or excessive numerical diffusion, is the dominant truncation error in the first-order upstream scheme.

13.3 The Courant-Friedrichs-Lewy criterion

The FTBS scheme is conditionally stable. In fact the criterion just derived from von Neumann stability analysis happens to coincide exactly with the Courant-Friedrichs-Lewy (CFL) criterion. *The CFL criterion states that, for a hyperbolic system, a necessary condition for stability is that the domain of dependence of the numerical solution should include the domain of dependence of the original partial differential equation.*

For the linear advection equation, the domain of dependence of the true solution at the point $(x_j, t^{(n)})$ is just the straight line of slope $1/u$ through $(x_j, t^{(n)})$ for $t \leq t^{(n)}$ in the (x,t) plane. For the FTBS scheme the numerical solution $\phi_j^{(n)}$ depends on $\phi_j^{(n-1)}$ and $\phi_{j-1}^{(n-1)}$. These in turn depend on $\phi_j^{(n-2)}$, $\phi_{j-1}^{(n-2)}$, and $\phi_{j-2}^{(n-2)}$, etc.

In the numerical solution information can propagate only from left to right, and can do so no faster than one grid length per time step. It is clear that the numerical domain of dependence will contain the physical domain of dependence provided that $1/u \geq \Delta t/\Delta x$, i.e. $0 \leq u\Delta t/\Delta x \leq 1$.

13.4 The CTCS scheme

As an alternative to the upstream scheme, consider the centred in time centred in space (CTCS) scheme

$$\frac{\phi_j^{(n+1)} - \phi_j^{(n-1)}}{2\Delta t} + u\,\frac{\phi_{j+1}^{(n)} - \phi_{j-1}^{(n)}}{2\Delta x} = 0.$$

This scheme is second-order accurate in space and time. It is a three-time-level formula since it involves values of ϕ at times $t^{(n-1)}$, $t^{(n)}$, and $t^{(n+1)}$. To start the integration values of ϕ are needed at times $t^{(0)}$ and $t^{(1)}$, but only $\phi_j^{(0)}$ is given as initial data, so another formula such as FTCS must be used to obtain $\phi_j^{(1)}$.

Von Neumann stability analysis leads to the following quadratic equation for the amplification factor:

$$A^2 + (2ic\sin k\Delta x)A - 1 = 0,$$

with roots

$$A = -ic\sin k\Delta x \pm \left[1 - (c\sin k\Delta x)^2\right]^{1/2}.$$

There are two cases to consider

- (i) $|c| < 1$

  Then $(c\sin k\Delta x)^2 < 1$, and

  $$|A|^2 = 1$$

  The solution is stable and, moreover, there is no damping error.

- (ii) $|c| > 1$

  Then, for some $k\Delta x$, $(c\sin k\Delta x)^2 > 1$, and

  $$|A|^2 = \left\{c\sin k\Delta x \pm \left[(c\sin k\Delta x)^2 - 1\right]^{1/2}\right\}^2$$

  One of the roots has $|A| > 1$, so the numerical solution is unstable.

13.5 Computational mode in the CTCS scheme

Because the CTCS scheme gives two roots for the amplification factor

$$A^{\pm} = -ic\sin k\Delta x \pm \left[1 - (c\sin k\Delta x)^2\right]^{1/2},$$

say, the general form of the numerical solution with wavenumber k is

$$\phi_j^{(n)} = \left[C_1\left(A^+\right)^n + C_2\left(A^-\right)^n\right]e^{ikj\Delta x}.$$

Assuming $|c| < 1$ so that $|A| = 1$, it is convenient to write

$$A^+ = e^{-i\alpha}, \qquad A^- = -e^{i\alpha},$$

where $\sin\alpha = c\sin k\Delta x$, $\cos\alpha = [1 - (c\sin k\Delta x)^2]^{1/2}$.

In the limit $\Delta x \to 0$, $\Delta t \to 0$ we find $\alpha \to uk\Delta t$, so that

$$\begin{aligned}\phi_j^{(n)} &= C_1 e^{ik(j\Delta x - un\Delta t)} + (-1)^n C_2 e^{ik(j\Delta x + un\Delta t)} \\ &= C_1 e^{ik(x-ut)} + (-1)^n C_2 e^{ik(x+ut)}.\end{aligned} \tag{22}$$

In this limit the first term becomes proportional to the exact solution $e^{ik(x-ut)}$ and is therefore called the *physical mode*. The second term does not correspond to any solution of the original equation; it is an artefact of the numerical method and is called the *computational mode*. Two characteristics of the computational mode for this scheme are (i) it oscillates in time from one step to the next (because of the $(-1)^n$ factor) and (ii) it propagates in the opposite direction to the physical solution (because of the $+un\Delta t$ in the exponential instead of $-un\Delta t$).

The solution $\phi_j^{(n+1)}$ depends on $\phi_{j\pm1}^{(n)}$ and $\phi_j^{(n-1)}$, but not on $\phi_j^{(n)}$. Except at the initial time, the solution is found on two sets of points that are not coupled: those with $n+j$ even, and those with $n+j$ odd. At any point x_j the solution oscillates between the two uncoupled solutions. To keep the amplitude of the computational mode small. it is necessary to couple the solutions on the two sets of alternating grid points. A common way to do this is to apply a time filter that strongly damps oscillations of period $2\Delta t$, e.g. the *Robert-Asselin filter*.

For a centred in time scheme

$$\phi^{(n+1)} = \phi^{(n-1)} + 2\Delta t(\text{other terms})$$

we replace $\phi^{(n-1)}$ by a filtered value

$$\overline{\phi}^{(n-1)} = \phi^{(n-1)} + \varepsilon\left(\phi^{(n)} - 2\phi^{(n-1)} + \overline{\phi}^{(n-2)}\right),$$

i.e., at each step compute

$$\phi^{(n+1)} = \overline{\phi}^{(n-1)} + 2\Delta t(\text{other terms})$$

then compute

$$\overline{\phi}^{(n)} = \phi^{(n)} + \varepsilon\left(\phi^{(n+1)} - 2\phi^{(n)} + \overline{\phi}^{(n-1)}\right),$$

for use in the next time step. This filter introduces some artificial damping of the physical mode, so ε should be kept small, e.g. $\varepsilon = 0.1$.

13.6 Amplitude and phase speed of the physical mode in the CTCS scheme

As already noted, $|A| = 1$ provided $|c| < 1$, so the amplitude of the physical mode neither grows nor decays. Therefore, amplitude errors will be small as long as the computational mode is weak.

For the physical mode $\phi_j^{(n)} \propto e^{ik(j\Delta x - \alpha n/k)} = e^{ik(j\Delta x - \alpha t/k\Delta t)}$, so its phase speed is

$$u_{\text{num}} = \frac{\alpha}{k\Delta t} = \frac{1}{k\Delta t}\sin^{-1}(c\sin k\Delta x).$$

Since $|c| < 1$ for stability, the phase speed is always underestimated. For a two-grid wave ($k\Delta x = \pi$) the phase speed is zero. For wavelengths much larger than the grid spacing, i.e. well resolved, $k\delta x$ is small so $u_{\text{num}} \approx u$. Thus, while the phase speed of the true solution is the same for all wavelengths (i.e. u) the phase speed in the numerical solution varies with wavelength. This leads to *numerical dispersion errors*.

13.7 Semi-Lagrangian advection

The CFL criterion tells us that any scheme for which $\phi_j^{(n+1)}$ depends only on its immediate neighbours $\phi_{j-1}^{(n)}$, $\phi_j^{(n)}$, $\phi_{j+1}^{(n)}$ at the previous time step will have a stability restriction on Δt. We can avoid the Δt restriction by explicitly constructing the numerical domain of dependence to contain the physical domain of dependence. This is what semi-Lagrangian advection schemes do.

Semi-Lagrangian schemes discretize the advective form of the advection equation

$$\frac{D\phi}{Dt} = 0$$

giving

$$\frac{\phi_j^{(n+1)} - \phi_{jD}^{(n)}}{\Delta t} = 0.$$

Here, the finite difference approximation to the time derivative is taken not at a fixed point in space, but along the trajectory of the air parcel that arrives at x_j at time $t^{(n+1)}$. Thus, the value $\phi_j^{(n+1)}$ is at grid point x_j, as before, but $\phi_{jD}^{(n)}$ must be evaluated at the the departure point of the trajectory. In order to implement a semi-Lagrangian scheme we must, therefore, (i) be able to determine the departure point of each trajectory, and (ii) evaluate $\phi_{jD}^{(n)}$ at the trajectory departure point.

For the linear advection equation the calculation of the departure point x_{jD} is trivial: it is simply $x_{jD} = x - u\Delta t$. For more complex situations, in which u varies in space or time, the departure point can only be determined approximately by an iterative method.

Once the departure point is known, the departure-point value $\phi_{jD}^{(n)}$ can be determined from the known grid point values $\phi_j^{(n)}$, $j = 0,\ 1,\ \ldots,\ N$ by an interpolation scheme. By using the appropriate interval as the basis for the interpolation, the CFL criterion can be satisfied for any size time step. In practice linear interpolation is found to lead to excessive numerical diffusion—indeed for $|c| < 1$ semi-Lagrangian with linear interpolation is equivalent to the first order upstream scheme for the linear advection equation. Various forms of cubic interpolation have been found to work well. They give very accurate phase speeds with only small damping errors.

Note that interpolation of a given order of accuracy leads to a semi-implicit advection scheme whose order of accuracy is one lower. For example, cubic interpolation, which is a fourth order accurate interpolation, leads to a third order accurate advection scheme.

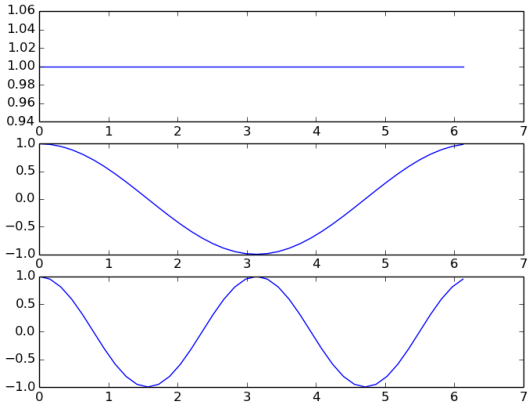

Figure 1: The first three cosines in the Fourier Cosine series.

14 The Spectral and Finite Element Methods

In the finite difference method we focused on the representation of a function by considering its Taylor Series in the neighborhood of a point x_j, and considering many such points. The finite element method and the spectral method are different conceptually in that they represent the function over the whole domain, using the concept of superposition (also called series expansion). For example,

$$f(x) = \sum_{k=0}^{N} \widehat{a}_k \ \psi_k(x), \tag{23}$$

where $\psi_k(x)$ are called trial, test, or basis functions.

The simplest example of this is the Fourier Series for periodic domains. If we consider the Fourier Cosine series then we can represent an even periodic function as truncated series of cosines,

$$\begin{aligned} f(x) &= \sum_{k=0}^{N} \widehat{a}_k \ cos(kx) \\ &= \widehat{a}_0 + \widehat{a}_1 cos(x) + \widehat{a}_2 cos(2x) + \widehat{a}_3 cos(3x) + ...\widehat{a}_N cos(Nx). \end{aligned} \tag{24}$$

For example, if $f(x) = cos(10x)$ then an exact representation of the function is possible if $\widehat{a}_{10} = 1$ and all the other coefficients are zero. Figure (14) shows the first three functions (for $k = 0, 1$, and 2) of the cosine series. The coefficients are found through a minimization technique called the Galerkin method. To represent a function using the Galerkin method use the expansion in Eq (23). Multiply each side of the equation by $\psi_l(x)$, for all $l = 0...N$, and integrating over the domain we see that,

$$\int_a^b f(x)\psi_l(x) \ dx = \int_a^b \sum_{k=0}^{N} \widehat{a}_k \ \psi_k(x)\psi_l(x) \ dx \tag{25}$$

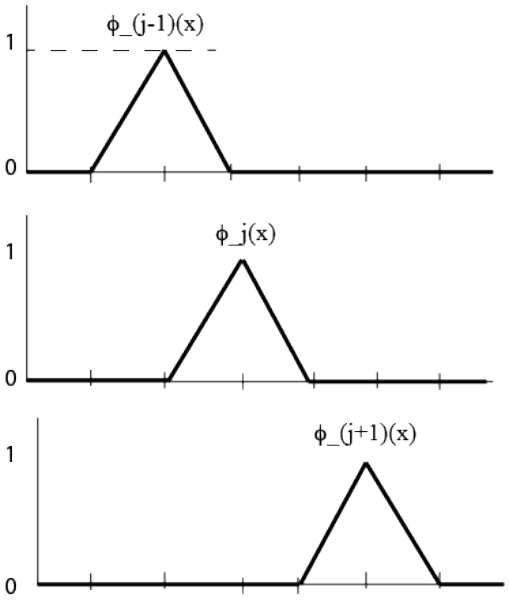

Figure 2: An example of hat or chapeau functions used in the finite element method.

For **spectral methods** the trial functions can be global orthonormal functions that have the following property,

$$\int_a^b \psi_k(x)\psi_l(x)dx = \delta_{kl}, \tag{26}$$

where δ_{kl} is the Kronecker delta function,

$$\delta_{kl} = \begin{cases} 1 & \text{when} \quad k = l \\ 0. & \text{otherwise} \end{cases} \tag{27}$$

Taking the integrals shows that the coefficients can be computed by the following equation,

$$\widehat{a}_l = \int_a^b f(x)\psi_l(x)dx. \tag{28}$$

This integral can be computed as you learned at the beginning of module. For common Fourier methods or Spherical Harmonics these can be computed efficiently.

Finite elements differ from spectral methods in that their trial functions are not generally global functions. The hat (or chapeau) functions are show in in Figure (14).

14.1 Linear Advection

Consider the finite element method for the linear advection equation Eq (20). As above, expand the unknown $\phi(x,t)$ as a series of hat functions,

$$\phi(x,t) = \sum_{n=1}^{N} \widehat{a}_n(t)\psi_n(x). \tag{29}$$

Substitute this into the linear advection equation to get,

$$\sum_{n=1}^{N} \frac{d\widehat{a}_n(t)}{dt}\psi_n(x) + u\sum_{n=1}^{N} \widehat{a}\frac{d\psi_n(x)}{dx}, \tag{30}$$

For each $l = 1, N$ multiply Eq (30) by $\psi_l(x)$ and integrate over the domain,

$$\int_0^1 \sum_{n=1}^{N} \frac{d\widehat{a}_n(t)}{dt}\psi_n(x)\ \psi_l(x)\ dx + u\int_0^1 \sum_{n=1}^{N} \widehat{a}_n\frac{d\psi_n(x)}{dx}\ \psi_l(x)dx, \tag{31}$$

The remaining task is to work out the integrals of the hat functions which have already been done in many different books on Finite Elements (we will use Duran, as recommended at the beginning of these notes).

$$\begin{aligned}
&\int_0^1 \psi_{j+1}\psi_j dx = \int_0^{\Delta x} \left(\frac{x}{\Delta x}\right)\left(\frac{\Delta x - x}{\Delta x}\right)\ dx = \frac{\Delta x}{6} \\
&\int_0^1 \psi_j\psi_j dx = 2\int_0^{\Delta x} \left(\frac{\Delta x - x}{\Delta x}\right)\ dx = \frac{2\Delta x}{3} \\
&\int_0^1 \frac{\partial\psi_{j+1}}{\partial x}\psi_j dx = -\int_0^1 \frac{\partial\psi_{j-1}}{\partial x}\psi_j dx = \int_0^{\Delta x} \left(\frac{1}{\Delta x}\right)\left(\frac{\Delta x - x}{\Delta x}\right)\ dx = \frac{1}{2}.
\end{aligned} \tag{32}$$

This leads to the following finite element method for the advection equation,

$$\frac{d}{dt}\left(\frac{\widehat{a}_{j+1} + 4\widehat{a}_j + \widehat{a}_{j-1}}{6}\right) + u\left(\frac{\widehat{a}_{j+1} - \widehat{a}_{j-1}}{2\Delta x}\right) = 0 \tag{33}$$

The method described by Eq (34) is an ordinary differential equation for the expansion coefficients. However, because we have defined the hat functions as having a value of 1 at the point j, they are also the nodal values! Therefore we can analyze this equation in the same way we have been analyzing finite difference methods. In fact, analysis of this scheme has shown that this particular finite element formulation of the advection equation is equivalent to a fourth order compact differencing scheme.

If one discretizes Eq (34) in time using standard methods such as Forward Euler we arrive at,

$$\left(\frac{\widehat{a}_{j+1}^{(n+1)} + 4\widehat{a}_j^{(n+1)} + \widehat{a}_{j-1}^{(n+1)}}{6}\right) = \left(\frac{\widehat{a}_{j+1}^{(n)} + 4\widehat{a}_j^{(n)} + \widehat{a}_{j-1}^{(n)}}{6}\right) - \frac{u\Delta t}{2\Delta x}\left(\widehat{a}_{j+1}^{(n)} - \widehat{a}_{j-1}^{(n)}\right). \tag{34}$$

Once again, the unknowns are on the left, the knowns on the right. This leads again to a tri-diagonal system of equations.

Another scheme to consider is Leap Frog. In this case then the stability analysis shows that for $\mu = u\Delta t/\Delta x < 1/\sqrt{3}$, the scheme is stable.

Because the finite element method minimizes the error in the function as defined by integrals, it can often be used to derive numerical schemes that automatically conserve quantities like mass and volume. This sometimes comes at the price of computational efficiency.