

A first course on Numerical Relativity

Luis Lehner (Perimeter Institute -Waterloo, Canada)
Vasileios Paschalidis (Princeton University, USA)
Frans Pretorius (Princeton University, USA)

IFT-UNESP
São Paulo, Brazil
March 28 - April 1, 2016

References

- Mitchell, A. R., and D. F. Griffiths, The Finite Difference Method in Partial Differential Equations, New York: Wiley (1980)
- Richtmeyer, R. D., and Morton, K. W., Difference Methods for Initial-Value Problems, New York: Interscience (1967)
- H.-O. Kreiss and J. Olinger, Methods for the Approximate Solution of Time Dependent Problems, GARP Publications Series No. 10, (1973)
- Gustatsson, B., H. Kreiss and J. Olinger, Time-dependent Problems and Difference Methods, New York: Wiley (1995)

Solution of Classical Field Equations Using Finite Difference Techniques

- 1. Solving the wave equation using finite difference techniques**

Preliminaries

- Classical field equations \equiv time dependent partial differential equations (PDEs)
- Can divide time-dependent PDEs into two broad classes:
 1. **Initial-value Problems (Cauchy Problems)**, spatial domain has no boundaries (either infinite or “closed”—e.g. “periodic boundary conditions”)
 2. **Initial-Boundary-Value Problems**, spatial domain *finite*, need to specify boundary conditions
- **Note:** Even if *physical* problem is really of type 1, finite computational resources \longrightarrow finite spatial domain \longrightarrow approximate as type 2; will hereafter loosely refer to either type as an IVP.
- *Working Definition:* **Initial Value Problem**
 - State of physical system arbitrarily (usually) specified at some initial time $t = t_0$.
 - Solution exists for $t \geq t_0$; uniquely determined by equations of motion (EOM) and boundary conditions (BCs).

Preliminaries

- Approximate solution of initial value problems using *any* numerical method, including finite differencing, will always involve three key steps
 1. Complete mathematical specification of system of PDEs, including boundary conditions and initial conditions
 2. Discretization of the system: replacement of continuous domain by discrete domain, and approximation of differential equations by algebraic equations for discrete unknowns
 3. Solution of discrete algebraic equations
- Will assume that the set of PDEs has a unique solution for given initial conditions and boundary conditions, and that the solution does not “blow up” in time, unless such blow up is expected from the physics
- Whenever this last condition holds for an initial value problem, we say that the problem is well posed
- Note that this is a non-trivial issue in general relativity, since there are in practice *many* distinct forms the PDEs can take for a given physical scenario (in principle infinitely many), and not all will be well-posed in general

Preliminaries

- Mathematical well-posedness
- Hyperbolicity
 - Weak
 - Strong
 - Strict
 - Symmetric
- Maximally Dissipative Boundary conditions

posedness and hyperbolicity presented here is of necessity brief and touches only on the main ideas, a more formal discussion can be found in the book by Kreiss and Lorenz [177] (see also the review paper of Reula [241]).

5.2 Well-posedness

Consider a system of partial differential equations of the form

$$\partial_t u = P(D)u , \tag{5.2.1}$$

where u is some n -dimensional vector-valued function of time and space, and $P(D)$ is an $n \times n$ matrix with components that depend smoothly on spatial derivative operators.⁴⁵ The *Cauchy* or *initial value* problem for such a system of equations corresponds to finding a solution $u(t, x)$ starting from some known initial data $u(t = 0, x)$.

A crucial property of a system of partial differential equations like the one considered above is that of *well-posedness*, by which we understand that the system is such that its solutions depend continuously on the initial data, or in other words, that small changes in the initial data will correspond to small changes in the solution. More formally, a system of partial differential equations is called well-posed if we can define a norm $\| \cdot \|$ such that

$$\|u(t, x)\| \leq k e^{\alpha t} \|u(0, x)\| , \tag{5.2.2}$$

with k and α constants that are independent of the initial data. That is, the norm of the solution can be bounded by the same exponential for all initial data.

Most systems of evolution equations we usually find in mathematical physics turn out to be well-posed, which explains why there has been some complacency in the numerical relativity community about this issue. However, it is in fact not difficult to find rather simple examples of evolution systems that are not well-posed. We will consider three such examples here. The easiest example is the inverse heat equation which can be expressed as

$$\partial_t u = -\partial_x^2 u . \tag{5.2.3}$$

Assume now that as initial data we take a Fourier mode $u(0, x) = e^{ikx}$. In that case the solution to the last equation can be easily found to be

$$u(x, t) = e^{k^2 t + ikx} . \tag{5.2.4}$$

We then see that the solution grows exponentially with time, with an exponent that depends on the frequency of the initial Fourier mode k . It is clear that by

⁴⁵One should not confuse the vectors we are considering here with vectors in the sense of differential geometry. A vector here only represents an ordered collection of independent variables.

increasing k we can increase the rate of growth arbitrarily, so the general solution can not be bounded by an exponential that is independent of the initial data. This also shows that given any arbitrary initial data, we can always add to it a small perturbation of the form ϵe^{ikx} , with $\epsilon \ll 1$ and $k \gg 1$, such that after a finite time the solution can be very different, so there is no continuity of the solutions with respect to the initial data.

A second example is the two-dimensional Laplace equation where one of the two dimensions is taken as representing “time”:

$$\partial_t^2 \phi = -\partial_x^2 \phi . \quad (5.2.5)$$

This equation can be trivially written in first order form by defining $u_1 := \partial_t \phi$ and $u_2 := \partial_x \phi$. We find

$$\partial_t u_1 = -\partial_x u_2 , \quad (5.2.6)$$

$$\partial_t u_2 = +\partial_x u_1 , \quad (5.2.7)$$

where the second equation simply states that partial derivatives of ϕ commute. Again, consider a Fourier mode as initial data. The solution is now found to be

$$\phi = \phi_0 e^{kt+ikx} , \quad u_1 = k\phi_0 e^{kt+ikx} , \quad u_2 = ik\phi_0 e^{kt+ikx} . \quad (5.2.8)$$

We again see that the solution grows exponentially with a rate that depends on the frequency of the initial data k , so it can not be bounded in a way that is independent of the initial data. This shows that the Laplace equation is ill-posed when seen as a Cauchy problem, and incidentally explains why numerical algorithms that attempt to solve the Laplace equation by giving data on one boundary and then “evolving” to the opposite boundary are bound to fail (numerical errors will explode exponentially as we march ahead).

The two examples above are rather artificial, as the inverse heat equation is unphysical and the Laplace equation is not really an evolution equation. Our third example of an ill-posed system is more closely related to the problem of the 3+1 evolution equations. Consider the simple system

$$\partial_t u_1 = \partial_x u_1 + \partial_x u_2 , \quad (5.2.9)$$

$$\partial_t u_2 = \partial_x u_2 . \quad (5.2.10)$$

This system can be rewritten in matrix notation as

$$\partial_t u = M \partial_x u , \quad (5.2.11)$$

with $u = (u_1, u_2)$ and M the matrix

$$M = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix} . \quad (5.2.12)$$

Again, consider the evolution of a single Fourier mode. The solution of the system of equations can then be easily shown to be

$$u_1 = (ikAt + B) e^{ik(t+x)}, \quad u_2 = Ae^{ik(t+x)}, \quad (5.2.13)$$

with A and B constants. Notice that u_2 is oscillatory in time, so it is clearly bounded. However, u_1 has both an oscillatory part and a linear growth in time with a coefficient that depends on the initial data. Again, it is impossible to bound the growth in u_1 with an exponential that is independent of the initial data, as for any time t we can always choose k large enough to surpass any such bound. The system is therefore ill-posed.

Systems of this last type in fact often appear in reformulations of the 3+1 evolution equations (particularly in the ADM formulation). The problem can be traced back to the form of the matrix M above. Such a matrix is called a *Jordan block* (of order 2 in this case), and has two identical real eigenvalues but can not be diagonalized.

In the following Section we will consider a special type of system of partial differential equations called *hyperbolic* that can be shown to be well-posed under very general conditions.

5.3 The concept of hyperbolicity

Consider a first order system of evolution equations of the form

$$\partial_t u + M^i \partial_i u = s(u), \quad (5.3.1)$$

where M^i are $n \times n$ matrices, with the index i running over the spatial dimensions, and $s(u)$ is a source vector that may depend on the u 's but not on their derivatives. In fact, if the source term is linear in the u 's we can show that the full system will be well-posed provided that the system without sources is well-posed. We will therefore ignore the source term from now on. Also, we will assume for the moment that the coefficients of the matrices M^i are constant.

There are several different ways of introducing the concept of *hyperbolicity* of a system of first order equations like (5.3.1).⁴⁶ Intuitively, the concept of hyperbolicity is associated with systems of evolution equations that behave as generalizations of the simple wave equation. Such systems are, first of all, well-posed, but they also should have the property of having a finite speed of propagation of signals, or in other words, they should have a finite past domain of dependence.

We will start by defining the notion of hyperbolicity based on the properties of the matrices M^i , also called the *characteristic matrices*. Consider an arbitrary unit vector n_i , and construct the matrix $P(n_i) := M^i n_i$, also known as the

⁴⁶One can in fact also define hyperbolicity for systems of second order equations (see for example [154, 155, 212]), but here we will limit ourselves to first order systems as we can always write the 3+1 evolution equations in this form.

principal symbol of the system of equations (one often finds that P is multiplied with the imaginary unit i , but here we will assume that the coefficients of the M^i are real so we will not need to do this). We then say that the system (5.3.1) is *strongly hyperbolic* if the principal symbol has real eigenvalues and a complete set of eigenvectors for all n_i . If, on the other hand, P has real eigenvalues for all n_i but does not have a complete set of eigenvectors then the system is said to be only *weakly hyperbolic* (an example of a weakly hyperbolic system is precisely the Jordan block considered in the previous Section). For a strongly hyperbolic system we can always find a positive definite Hermitian (*i.e.* symmetric in the purely real case) matrix $H(n_i)$ such that

$$HP - P^T H^T = HP - P^T H = 0, \quad (5.3.2)$$

where the superindex T represents the transposed matrix. In other words, the new matrix HP is also symmetric, and H is called the *symmetrizer*. The symmetrizer is in fact easy to find. By definition, if the system is strongly hyperbolic the symbol P will have a complete set of eigenvectors e_a such that (here the index a runs over the dimensions of the space of solutions u)

$$Pe_a = \lambda_a e_a, \quad (5.3.3)$$

with λ_a the corresponding eigenvalues. Define now R as the matrix of column eigenvectors. The matrix R can clearly be inverted since all the eigenvectors are linearly independent. The symmetrizer is then given by

$$H = (R^{-1})^T R^{-1}, \quad (5.3.4)$$

which is clearly Hermitian and positive definite. To see that HP is indeed symmetric notice first that

$$R^{-1}PR = \Lambda, \quad (5.3.5)$$

with $\Lambda = \text{diag}(\lambda_a)$ (this is just a similarity transformation of P into the basis of its own eigenvectors). We can then easily see that

$$HP = (R^{-1})^T R^{-1}P = (R^{-1})^T \Lambda R^{-1}. \quad (5.3.6)$$

But Λ is diagonal so that $\Lambda^T = \Lambda$, which immediately implies that $(R^{-1})^T \Lambda R^{-1}$ is symmetric. Of course, since the eigenvectors e_a are only defined up to an arbitrary scale factor and are therefore not unique, the matrix R and the symmetrizer H are not unique either.

We furthermore say that the system of equations is *symmetric hyperbolic* if all the M^i are symmetric, or more generally if the symmetrizer H is independent of n_i . Symmetric hyperbolic systems are therefore also strongly hyperbolic, but not all strongly hyperbolic systems are symmetric. Notice also that in the case of one spatial dimension any strongly hyperbolic system can be symmetrized, so the distinction between symmetric and strongly hyperbolic systems does not

arise. We can also define a *strictly hyperbolic* system as one for which the eigenvalues of the principal symbol P are not only real but are also distinct for all n_i . Of course, this immediately implies that the symbol can be diagonalized, so strictly hyperbolic systems are automatically strongly hyperbolic. This last concept, however, is of little use in physics where we often find that the eigenvalues of P are degenerate, particularly in the case of many dimensions.

The importance of the symmetrizer H is related to the fact that we can use it to construct an inner product and norm for the solutions of the differential equation in the following way

$$\langle u, v \rangle := u^\dagger H v, \quad (5.3.7)$$

$$\|u\|^2 := \langle u, u \rangle = u^\dagger H u, \quad (5.3.8)$$

where u^\dagger is the adjunct of u , *i.e.* its complex-conjugate transpose (we will allow complex solutions in order to use Fourier modes in the analysis). In geometric terms the matrix H plays the role of the metric tensor in the space of solutions. The norm defined above is usually called an *energy norm* since in some simple cases it coincides with the physical energy.

We can now use the evolution equations to estimate the growth in the energy norm. Consider a Fourier mode of the form

$$u(x, t) = \tilde{u}(t) e^{ik\bar{x} \cdot \bar{n}}. \quad (5.3.9)$$

We will then have

$$\begin{aligned} \partial_t \|u\|^2 &= \partial_t (u^\dagger H u) = \partial_t (u^\dagger) H u + u^\dagger H \partial_t (u) \\ &= ik\tilde{u}^T P^T H \tilde{u} - ik\tilde{u}^T H P \tilde{u} \\ &= ik\tilde{u}^T (P^T H - H P) \tilde{u} = 0, \end{aligned} \quad (5.3.10)$$

where on the second line we have used the evolution equation (assuming $s = 0$). We then see that the energy norm remains constant in time. This shows that strongly and symmetric hyperbolic systems are well-posed. We can in fact show that hyperbolicity and the existence of a conserved energy norm are equivalent, so instead of analyzing the principal symbol P we can look directly for the existence of a conserved energy to show that a system is hyperbolic. Notice that for symmetric hyperbolic systems the energy norm will be independent of the vector n_i , but for systems that are only strongly hyperbolic the norm will in general depend on n_i .

Now, for a strongly hyperbolic system we have by definition a complete set of eigenvectors and we can construct the matrix of eigenvectors R . We will use this matrix to define the *eigenfunctions* w_i (also called *eigenfields*) as

$$u = R w \quad \Rightarrow \quad w = R^{-1} u. \quad (5.3.11)$$

Notice that, just as was the case with the eigenvectors, the eigenfields are only defined up to an arbitrary scale factor. Consider now the case of a single spatial dimension x . By multiplying equation (5.3.1) with R^{-1} on the left we find that

$$\partial_t w + \Lambda \partial_x w = 0, \quad (5.3.12)$$

so that the evolution equations for the eigenfields decouple. We then have a set of independent advection equations, each with a speed of propagation given by the corresponding eigenvalue λ_a . This is the mathematical expression of the notion that associates a hyperbolic system with having independent “wave fronts” propagating at (possibly different) finite speeds. Of course, in the multidimensional case the full system will generally not decouple even for symmetric hyperbolic systems, as the eigenfunctions will depend on the vector n_i .

We can in fact use the eigenfunctions also to study the hyperbolicity of a system; the idea here would be to construct a complete set of linearly independent eigenfunctions w_a that evolve via simple advection equations starting from the original variables u_a . If this is possible then the system will be strongly hyperbolic. For systems with a large number of variables this method is often simpler than constructing the eigenvectors of the principal symbol directly, as finding eigenfunctions can often be done by inspection (this is in fact the method we will use in the following Sections to study the hyperbolicity of the different 3+1 evolution systems).

Up until now we have assumed that the characteristic matrices M^i have constant coefficients, and also that the source term $s(u)$ vanishes. In the more general case when $s(u) \neq 0$ and $M^i = M^i(t, x, u)$ we can still define hyperbolicity in the same way by linearizing around a background solution $\hat{u}(t, x)$ and considering the local form of the matrices M^i , and we can also show that strong and symmetric hyperbolicity implies well-posedness. The main difference is that now we can only show that solutions exist locally in time, as after a finite time singularities in the solution may develop (*e.g.* shock waves in hydrodynamics, or spacetime singularities in relativity). Also, the energy norm does not remain constant in time but rather grows at a rate that can be bounded independently of the initial data. A particularly important sub-case is that of *quasi-linear* systems of equations where we have two different sets of variables u and v such that derivatives in both space and time of the u 's can always be expressed as (possibly non-linear) combinations of v 's, and the v 's evolve through equations of the form $\partial_t v + M^i(u) \partial_i v = s(u, v)$, with the matrices M^i functions only of the u 's. In such a case we can bring the u 's freely in and out of derivatives in the evolution equations of the v without changing the principal part by replacing all derivatives of u 's in terms of v 's, and all the theory presented here can be applied directly. As we will see later, the Einstein field equations have precisely this property, with the u 's representing the metric coefficients (lapse, shift and spatial metric) and the v 's representing both components of the extrinsic curvature and spatial derivatives of the metric.

First order systems of equations of type (5.3.1) are often written instead as

$$\partial_t u + \partial_i F^i(u) = s(u), \quad (5.3.13)$$

where F^i are vector valued functions of the u 's (and possibly the spacetime

better alternative would be to evolve the quantity $\tilde{V}^i := \tilde{\Gamma}^i - 8 \tilde{\gamma}^{ik} \partial_k \phi$, instead of the $\tilde{\Gamma}^i$, because it propagates along time lines and would therefore require no boundary condition in the case of zero shift). Far worse, however, would be to impose a radiative boundary condition on the spatial derivatives of the $\tilde{\gamma}_{ij}$, since in that case we would be giving boundary data for outgoing fields as well. But as already mentioned, this is not done when we work with evolution equations that are second order in space.

5.9.2 Maximally dissipative boundary conditions

Let us now go back to the issue of finding well-posed boundary conditions for a symmetric hyperbolic system of equations. The restriction to symmetric hyperbolic systems is important in order to be able to prove well-posedness. For systems that are only strongly hyperbolic but not symmetric, like BSSNOK, no rigorous results exist about the well-posedness of the initial-boundary value problem. We then start by considering, as before, an evolution system of the form

$$\partial_t u + M^i \partial_i u = 0, \quad (5.9.11)$$

where the matrices M^i are constant, and the domain of dependence of the solution is restricted to the region $\vec{x} \in \Omega$. Let us now construct the principal symbol $P(n_i) := M^i n_i$, with n_i an arbitrary unit vector. We will assume that the system is symmetric hyperbolic, which implies that there exists a symmetrizer H , independent of the vector n_i , that is a Hermitian matrix such that $HP - P^T H = 0$. Consider now the energy norm

$$E(t) = \int_{\Omega} u^\dagger H u \, dV. \quad (5.9.12)$$

Taking a time derivative of this energy we find

$$\begin{aligned} \frac{dE}{dt} &= - \int_{\Omega} [(\partial_i u^\dagger) M^{iT} H u + u^\dagger H M^i (\partial_i u)] \, dV \\ &= - \int_{\Omega} [(\partial_i u^\dagger) H M^i u + u^\dagger H M^i (\partial_i u)] \, dV \\ &= - \int_{\Omega} \partial_i (u^\dagger H M^i u) \, dV, \end{aligned} \quad (5.9.13)$$

where in the second line we used the fact that H is the *same symmetrizer* for all n_i , which in particular means that all three matrices $H M^i$ are symmetric. This is precisely the place where the assumption that we have a symmetric hyperbolic system becomes essential. Using now the divergence theorem we finally find

$$\frac{dE}{dt} = - \int_{\partial\Omega} (u^\dagger H M^i u) n_i \, dA = - \int_{\partial\Omega} (u^\dagger H P(\vec{n}) u) \, dA, \quad (5.9.14)$$

where $\partial\Omega$ is the boundary of Ω , \vec{n} and dA are the normal vector to the boundary and its corresponding area element, and $P(\vec{n}) = M^i n_i$ is the symbol associated

with \vec{n} . In contrast to what we have done before, we will now not assume that the surface integral above vanishes. Instead, we now make use of equation (5.3.6):

$$HP = (R^{-1})^T R^{-1}P = (R^{-1})^T \Lambda R^{-1}, \tag{5.9.15}$$

with R the matrix of column eigenvectors of $P(\vec{n})$ and $\Lambda = \text{diag}(\lambda_i)$ the matrix of corresponding eigenvalues. We can therefore rewrite the change in the energy norm as

$$\frac{dE}{dt} = - \int_{\partial\Omega} (u^\dagger (R^{-1})^T \Lambda R^{-1} u) dA = - \int_{\partial\Omega} (w^\dagger \Lambda w) dA, \tag{5.9.16}$$

where $w := R^{-1}u$ are the eigenfields. Let w_+, w_-, w_0 now denote the eigenfields corresponding to eigenvalues of $P(\vec{n})$ that are positive, negative and zero respectively, *i.e.* eigenfields that propagate outward, inward, and tangential to the boundary respectively.⁵⁷ We then find

$$\begin{aligned} \frac{dE}{dt} &= - \int_{\partial\Omega} (w_+^\dagger \Lambda_+ w_+) dA - \int_{\partial\Omega} (w_-^\dagger \Lambda_- w_-) dA \\ &= \int_{\partial\Omega} (w_-^\dagger |\Lambda_-| w_-) dA - \int_{\partial\Omega} (w_+^\dagger |\Lambda_+| w_+) dA, \end{aligned} \tag{5.9.17}$$

with Λ_+ and Λ_- the sub-matrices of positive and negative eigenvalues. We clearly see that the first term in the last expression is always positive, while the second term is always negative. This shows that outward propagating fields (those with positive speed) reduce the energy norm since they are leaving the region Ω , while inward propagating modes (those with negative speed) increase it since they are coming in from the outside.

Assume that we now impose a boundary condition of the following form

$$w_-|_{\partial\Omega} = S w_+|_{\partial\Omega}, \tag{5.9.18}$$

with S some matrix that relates incoming fields at the boundary to outgoing ones. We then have

$$\begin{aligned} \frac{dE}{dt} &= \int_{\partial\Omega} (w_+^\dagger S^T |\Lambda_-| S w_+) dA - \int_{\partial\Omega} (w_+^\dagger |\Lambda_+| w_+) dA \\ &= \int_{\partial\Omega} [w_+^\dagger (S^T |\Lambda_-| S - |\Lambda_+|) w_+] dA. \end{aligned} \tag{5.9.19}$$

From this we clearly see that if we take S to be “small enough” in the sense that $w_+^\dagger S^T |\Lambda_-| S w_+ \leq w_+^\dagger |\Lambda_+| w_+$, then the energy norm will not increase

⁵⁷We should be careful with the interpretation of w_+ and w_- , because in many references we find their meaning reversed. This comes from the fact that we often find the evolution system written as $\partial_t u = M^i \partial_i u$ instead of the form $\partial_t u + M^i \partial_i u = 0$ used here, which of course reverses the signs of all the matrices and in particular of the matrix of eigenvalues Λ .

with time and the full system including the boundaries will remain well-posed. Boundary conditions of this form are known as *maximally dissipative* [185]. The particular case $S = 0$ corresponds to saying that the incoming fields vanish, and this results in a Sommerfeld-type boundary condition. This might seem the most natural condition, but it is in fact not always a good idea, as we might find that in order to reproduce the physics correctly (*e.g.* to satisfy the constraints) we might need to have some non-zero incoming fields at the boundary.

We can in fact generalize the above boundary condition somewhat to allow for free data to enter the domain. We can then take a boundary condition of the form

$$w_-|_{\partial\Omega} = S w_+|_{\partial\Omega} + g(t) , \quad (5.9.20)$$

where $g(t)$ is some function of time that represents incoming radiation at the boundary, and where as before we ask for S to be small. In this case we are allowing the energy norm to grow with time, but in a way that is bounded by the integral of $|g(t)|$ over the boundary, so the system remains well-posed. In the same way we can also allow for the presence of source terms on the right hand side of the evolution system (5.9.11).

As a simple example of the above results we will consider again the wave equation in spherical symmetry

$$\partial_t^2 \varphi - v^2 \left(\partial_r^2 \varphi + \frac{2}{r} \partial_r \varphi \right) = 0 . \quad (5.9.21)$$

Introducing the first order variables $\Pi := \partial_t \varphi$ and $\Psi := v \partial_r \varphi$, the wave equation can be reduced to the system

$$\partial_t \varphi = \Pi , \quad (5.9.22)$$

$$\partial_t \Pi = v \partial_r \Psi + \frac{2v}{r} \Pi , \quad (5.9.23)$$

$$\partial_t \Psi = v \partial_r \Pi . \quad (5.9.24)$$

The system above is clearly symmetric hyperbolic, with eigenspeeds $\{0, \pm v\}$ and corresponding eigenfields

$$w_0 = \varphi , \quad w_{\pm} = \Pi \mp \Psi . \quad (5.9.25)$$

Let us consider now the maximally dissipative boundary conditions at a sphere of radius $r = R$. These boundary conditions have the form

$$w_- = S w_+ + g(t) . \quad (5.9.26)$$

The requirement for S to be small now reduces simply to $S^2 \leq 1$. We will consider three particular cases:

- $S = -1$. This implies $\Pi + \Psi = -(\Pi - \Psi) + g(t)$, or in other words $\Pi = g(t)/2$. Since $\Pi = \partial_t \varphi$, this boundary condition fixes the evolution of φ at the boundary, so it corresponds to a boundary condition of Dirichlet type. The particular case $g = 0$ results in a standard reflective boundary condition, where the sign of φ changes as it reflects from the boundary.
- $S = +1$. This now implies $\Psi = g(t)/2$, which fixes the evolution of the spatial derivative of the wave function φ and corresponds to a boundary condition of Neumann type. Again, the case $g = 0$ corresponds to reflection, but preserving the sign of φ .
- $S = 0$. In this case we have $\Pi + \Psi = g(t)$, or in terms of the wave function $\partial_t \varphi + v \partial_r \varphi = g(t)$. This is therefore a boundary condition of Sommerfeld type.

From the expressions for dE/dt given above, it is easy to see that the choices $S = \pm 1$ with $g = 0$ imply that the energy norm is preserved (all the energy that leaves the domain through the outgoing modes comes back in through the incoming modes), so the wave is reflected at the boundary. Notice also that in the Sommerfeld case $S = 0$ we have not quite recovered the radiative boundary condition of the previous Section. But this is not a serious problem and only reflects the fact that we have excluded the source terms from all of our analysis. However, it does show that in many cases we need to consider a more general boundary condition of the form

$$w_-|_{\partial\Omega} = (S_+ w_+ + S_0 w_0)|_{\partial\Omega} + g(t) . \quad (5.9.27)$$

5.9.3 Constraint preserving boundary conditions

As discussed in the previous Section, the use of maximally dissipative boundary conditions for a symmetric hyperbolic system is crucial if we wish to have a well-posed initial-boundary value problem. However, this is not enough in the case of the 3+1 evolution equations since well-posed boundary conditions can still introduce a violation of the constraints that will then propagate into the computational domain at essentially the speed of light (the specific speed will depend on the form of the evolution equations used). We then have to worry about finding boundary conditions that are not only well-posed, but at the same time are compatible with the constraints. In a seminal work [133], Friedrich and Nagy have shown for the first time that it is possible to find a well-posed initial-boundary value formulation for the Einstein field equations that preserves the constraints. Their formulation, however, is based on the use of an orthonormal tetrad and takes as dynamical variables the components of the connection and the Weyl curvature tensor, so it is very different from most 3+1 formulations that evolve the metric and extrinsic curvature directly. It is therefore not clear how to apply their results to these standard “metric” formulations.

In the past few years, there have been numerous investigations related to the issue of finding well-posed constraint preserving boundary conditions [40, 61, 87, 88, 89, 137, 138, 154, 174, 191, 251, 278, 279, 280]. Here we will just present the

Preliminaries

- Initial value problem for ordinary differential equations (ODEs)
- Assume we have a coupled system of ODEs (u = vector of unknown variables)

$$\frac{d^2 u}{dt^2} = F(u), \quad u(0) = u_0, \quad u'(0) = v_0$$

- Find $u(t)$?

- Recast system as $\frac{du}{dt} = v, \quad \frac{dv}{dt} = F(u),$

$$u(0) = u_0, \quad v(0) = v_0$$

Preliminaries

$$\frac{du}{dt} = v, \quad \frac{dv}{dt} = F(u)$$

- **Euler**: simple first-order accurate (in Δt) integration method

- Denoting $u^n = u(n \cdot \Delta t)$, approximate derivative as

$$\left(\frac{du}{dt}\right)^n = \frac{u^{n+1} - u^n}{\Delta t}$$

- Then: $u^{n+1} = u^n + \Delta t v^n$ n+1
 $v^{n+1} = v^n + \Delta t F(u^n)$ u,v
n

Preliminaries

$$\frac{du}{dt} = v, \quad \frac{dv}{dt} = F(u)$$

- **Leap frog**: simple second-order accurate (in Δt) integration method
- Approximate derivative as

$$\left(\frac{du}{dt}\right)^n = \frac{u^{n+1} - u^{n-1}}{2 \Delta t}$$

- Then: $u^{n+1} = u^{n-1} + 2 \Delta t v^n$ n+1
- $v^{n+1} = v^{n-1} + 2 \Delta t F(u^n)$ u,v
n
- $v^n = v^{n-1} + \Delta t F(u^n)$ u,v
n-1

Preliminaries

- In general we will have to solve $\frac{du}{dt} = F(u)$, $u(0) = u_0$
- Leapfrog simple but, there are better, more accurate (higher-order) and more stable numerical schemes
- Examples: Runge-Kutta (RK) 2nd, 3rd, 4th order

RK3

$$k_1 = \Delta t F(u^n)$$

$$k_2 = \Delta t F(u^n + k_1/3)$$

$$k_3 = \Delta t F(u^n + 2k_2/3)$$

$$u^{n+1} = u^n + \frac{k_1}{4} + \frac{3k_3}{4}$$

RK4

$$k_1 = \Delta t F(u^n)$$

$$k_2 = \Delta t F(u^n + k_1/2)$$

$$k_3 = \Delta t F(u^n + k_2/2)$$

$$k_4 = \Delta t F(u^n + k_3)$$

$$u^{n+1} = u^n + \frac{k_1}{6} + \frac{k_2}{3} + \frac{k_3}{3} + \frac{k_4}{6}$$

Preliminaries

- Solving the initial value problem for partial differential equations (PDEs)

- In general we will have to solve

$$\frac{\partial u}{\partial t} = F(u, \partial_i u, \partial_i \partial_j u, \dots), \quad u(0, x^i) = u_0, \quad i = 1, 2, 3$$

- **The concept of the method of lines**

- Discretize the spatial derivatives

- Find a numerical approximation F_N to the function F

- Treat $\frac{\partial u}{\partial t} = F_N$ as a large number of ODEs (one for each spatial cell) and use your favorite ODE solver.

Why Finite Differencing?

- There are several general approaches to the numerical solution of time dependent PDEs, including
 1. Finite differences
 2. Finite volume
 3. Finite elements
 4. Spectral
- Finite difference (FD) methods are particularly appropriate when the solution is expected to be smooth “(infinitely differentiable”) given that the initial data is smooth
- This is the case for many classical field theories including those for a scalar (linear/nonlinear Klein Gordon), vector (electromagnetism [Maxwell]), rank-2 symmetric tensor (general relativity [Einstein])
- In cases where solutions do *not* remain smooth, even if the initial data is—as happens in compressible hydrodynamics, for example, where shocks can form -- the finite volume approach is the method of choice (Wednesday)

Why Finite Differencing?

- Accessibility: Requires a minimum of mathematical background: if you're mathematically mature enough to understand the nature of the PDEs you need to solve, you're mathematically mature enough to understand finite differencing
- Flexibility: Technique can be used for essentially any system of PDEs that has smooth solutions, irrespective of
 - Number of dependent variables (unknown functions)
 - Number of independent variables (a.k.a. "dimensionality" of the system: nomenclature "1-D" means dependence on one spatial dimension plus time, "2-D", "3-D" similarly mean dependence on two/three dimensions, plus time, respectively)
 - Nonlinearity
 - Form of equations: technique does not require that the system of equations has any particular/special form (contrast with finite volume methods where one generally wants to cast the equations in so-called conservation-law form)

Why Finite Differencing?

- Error analysis:
 - Mathematically rigorous: Quite difficult
 - Practical/empirical: Extremely straightforward—basic principle is to compute multiple solutions using same initial data and problem parameters, but differing fundamental discretization scales. Comparison of solutions provides direct estimate of error in solutions
- Adaptivity: Can combine basic method with changes in
 - Local scale of discretization
 - Order of approximation

in order to maximize increase in solution accuracy as a function of computational work invested (e.g. adaptive mesh refinement, week 3)
- Parallelization: Due to “locality of influence” in finite difference schemes, it is relatively easy to write FD codes than run efficiently on large distributed memory computer clusters having 1000s or cores (these days 10,000s or even 100,000s!)

Why Finite Differencing?

- Sufficiency: FD techniques are often sufficient to generate solutions of acceptable accuracy, again assuming that solutions are smooth
 - Will usually not be the most efficient and/or accurate among possible approaches, but when one is looking for a solution for the first time (science vs engineering/technology), such considerations are often not very important
- Now proceed to illustration of finite difference technique through the solution of the simple and familiar 1-D wave equation

1. Mathematical Formulation

The 1-D Wave Equation

- Consider the following initial value (Cauchy) problem for the scalar function $\phi(t, x)$

$$\phi_{tt} = c^2 \phi_{xx}, \quad -\infty \leq x \leq \infty, \quad t \geq 0 \quad (1)$$

$$\phi(0, x) = \phi_0(x) \quad (2)$$

$$\phi_t(0, x) = \Pi_0(x) \quad (3)$$

where c is a positive constant, we have adopted the subscript notation for partial differentiation, e.g. $\phi_{tt} \equiv \partial^2 \phi / \partial t^2$, and we wish to determine $\phi(t, x)$ in the solution domain from the initial conditions (2–3) and the governing equation (1)

- Note the following:
 - Since the spatial domain is unbounded, there are *no* boundary conditions
 - Since the equation is second order in time, two functions-worth of initial data must be specified: the initial scalar field profile, $\phi_0(x)$, and the initial time derivative, $\Pi_0(x)$
 - This system is well posed, and if the initial conditions $\phi_0(x)$ and $\Pi_0(x)$ are smooth—which we will hereafter assume—so is the complete solution $\phi(t, x)$

The 1-D Wave Equation

- Eqn. (1) is a *hyperbolic* PDE, and as such, its solutions generically describe the propagation of disturbances at some finite speed(s), which in this case is c
- Without loss of generality, we can assume that we have adopted units in which this speed satisfies $c = 1$. Our problem then becomes

$$\phi_{tt} = \phi_{xx}, \quad -\infty \leq x \leq \infty, \quad t \geq 0 \quad (4)$$

$$\phi(0, x) = \phi_0(x) \quad (5)$$

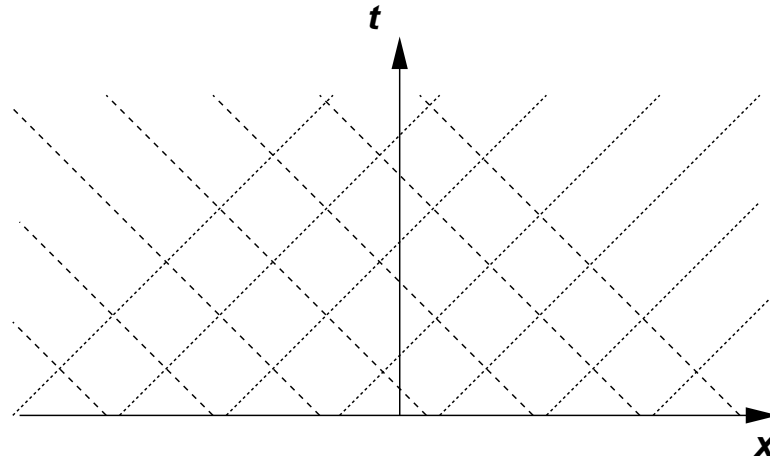
$$\phi_t(0, x) = \Pi_0(x) \quad (6)$$

- In the study of the solutions of hyperbolic PDEs, using either closed form (preferred to “analytic”) or numerical approaches, the concept of characteristic is crucial
- Loosely, in a spacetime diagram, characteristics are the lines/surfaces along which information/signals propagate(s).

The 1-D Wave Equation

----- : "left-directed" characteristics, $x + t = \text{constant}$, $l(x + t)$

----- : "right-directed" characteristics, $x - t = \text{constant}$, $r(x - t)$



- General solution of (4) is a superposition of an arbitrary *left-moving* profile ($v = -c = -1$), and an arbitrary *right-moving* profile ($v = +c = +1$); i.e.

$$\phi(t, x) = \ell(x + t) + r(x - t) \quad (7)$$

where

ℓ : constant along "left-directed" characteristics

r : constant along "right-directed" characteristics

The 1-D Wave Equation

- Observation provides alternative way of specifying initial values—often convenient in practice
- Rather than specifying $u(x, 0)$ and $u_t(x, 0)$ directly, specify *initial* left-moving and right-moving parts of the solution, $\ell(x)$ and $r(x)$
- Specifically, set

$$\phi(x, 0) = \ell(x) + r(x) \quad (8)$$

$$\phi_t(x, 0) = \ell'(x) - r'(x) \equiv \frac{d\ell}{dx}(x) - \frac{dr}{dx}(x) \quad (9)$$

- For illustrative purposes will frequently take profile functions $\phi_0(x)$, $\ell(x)$, $r(x)$ to be “gaussians”, e.g.

$$\phi_0(x) = A \exp \left[- ((x - x_0) / \delta)^2 \right] \quad (10)$$

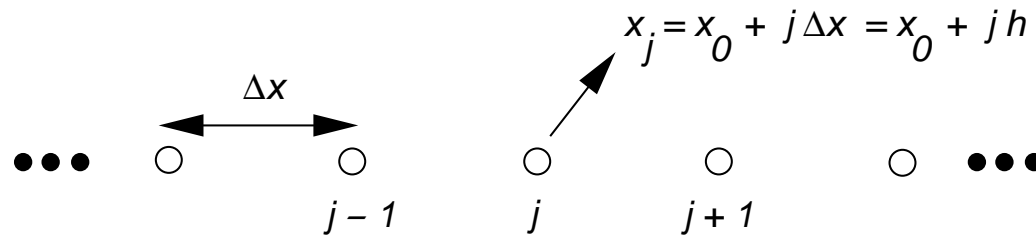
where A , x_0 and δ are viewed as adjustable parameters that control the overall size/height of the profile (A), its centre point (x_0) and its effective width (δ)

2. Discretization

Deriving Finite Difference Formulae

- Essence of finite-difference approximation of a PDE:
 - Replacement of the continuum by a discrete lattice of grid points
 - Replacement of derivatives/differential operators by finite-difference expressions
- Finite-difference expressions (finite-difference quotients) approximate the derivatives of functions at grid points, using the grid values themselves. All operators and expressions needed here can easily be worked out using Taylor series techniques.
- Example: Consider task of approximating the first derivative $u_x(x)$ of a function $u(x)$, given a discrete set of values $u_j \equiv u(jh)$

Deriving Finite Difference Formulae



- One-dimensional, uniform finite difference mesh.
- Note that the spacing, $\Delta x = h$, between adjacent mesh points is *constant*.
- Will tacitly assume that the origin, x_0 , of coordinate system is $x_0 = 0$.

Deriving Finite Difference Formulae

- Given the three values $u(x_j - h)$, $u(x_j)$ and $u(x_j + h)$, denoted u_{j-1} , u_j , and u_{j+1} respectively, can compute an $O(h^2)$ approximation to $u_x(x_j) \equiv (u_x)_j$ as follows
- Taylor expanding, have

$$u_{j-1} = u_j - h(u_x)_j + \frac{1}{2}h^2(u_{xx})_j - \frac{1}{6}h^3(u_{xxx})_j + \frac{1}{24}h^4(u_{xxxx})_j + O(h^5)$$

$$u_j = u_j$$

$$u_{j+1} = u_j + h(u_x)_j + \frac{1}{2}h^2(u_{xx})_j + \frac{1}{6}h^3(u_{xxx})_j + \frac{1}{24}h^4(u_{xxxx})_j + O(h^5)$$

- Now seek a linear combination of u_{j-1} , u_j , and u_{j+1} which yields $(u_x)_j$ to $O(h^2)$ accuracy, i.e. we seek c_- , c_0 and c_+ such that

$$c_- u_{j-1} + c_0 u_j + c_+ u_{j+1} = (u_x)_j + O(h^2)$$

Deriving Finite Difference Formulae

- Results in a system of three linear equations for u_{j-1} , u_j , and u_{j+1} :

$$\begin{aligned}c_- + c_0 + c_+ &= 0 \\-hc_- + hc_+ &= 1 \\\frac{1}{2}h^2c_- + \frac{1}{2}h^2c_+ &= 0\end{aligned}$$

which has the solution

$$\begin{aligned}c_- &= -\frac{1}{2h} \\c_0 &= 0 \\c_+ &= +\frac{1}{2h}\end{aligned}$$

- Thus, $O(h^2)$ FDA (finite difference approximation) for the first derivative is

$$\frac{u(x+h) - u(x-h)}{2h} = u_x(x) + O(h^2) \quad (11)$$

Deriving Finite Difference Formulae

- May not be obvious *a priori*, that the truncation error of approximation is $O(h^2)$
- Naive consideration of the number of terms in the Taylor series expansion which can be eliminated using 2 values (namely $u(x+h)$ and $u(x-h)$) suggests that the error might be $O(h)$.
- Fact that the $O(h)$ term “drops out” a consequence of the *symmetry*, or *centering* of the stencil: common theme in such FDA, called *centred* difference approximations
- Using same technique, can easily generate $O(h^2)$ expression for the *second* derivative, which uses the same difference stencil as the above approximation for the first derivative.

$$\frac{u(x+h) - 2u(x) + u(x-h)}{h^2} = u_{xx}(x) + O(h^2) \quad (12)$$

- *Exercise:* Compute the precise form of the $O(h^2)$ terms in expressions (11) and (12).

Sample FDA for the 1-D Wave Equation

- Let us consider the 1-D wave equation again, but this time on the finite spatial domain, $0 \leq x \leq 1$, where we will prescribe fixed (Dirichlet) boundary conditions
- Then we wish to solve

$$\phi_{tt} = \phi_{xx} \quad (c = 1) \quad 0 \leq x \leq 1, \quad t \geq 0 \quad (13)$$

$$\phi(0, x) = \phi_0(x)$$

$$\phi_t(0, x) = \Pi_0(x)$$

$$\phi(t, 0) = \phi(t, 1) = 0 \quad (14)$$

- We will again require that the initial data functions, $\phi_0(x)$ and $\Pi_0(x)$ be smooth
- Moreover, in order to ensure a smooth solution everywhere, the initial values must be compatible with the boundary conditions, i.e.

$$\phi_0(0) = \phi_0(1) = \Pi_0(0) = \Pi_0(1) = 0 \quad (15)$$

Sample FDA for the 1-D Wave Equation

- As always, we begin the discretization process by replacing the continuum solution domain with a finite difference mesh, whose typical element (point/event) we will denote by (x_j, t^n) :

$$t^n \equiv n \Delta t, \quad n = 0, 1, 2, \dots$$

$$x_j \equiv (j - 1) \Delta x, \quad j = 1, 2, \dots, J$$

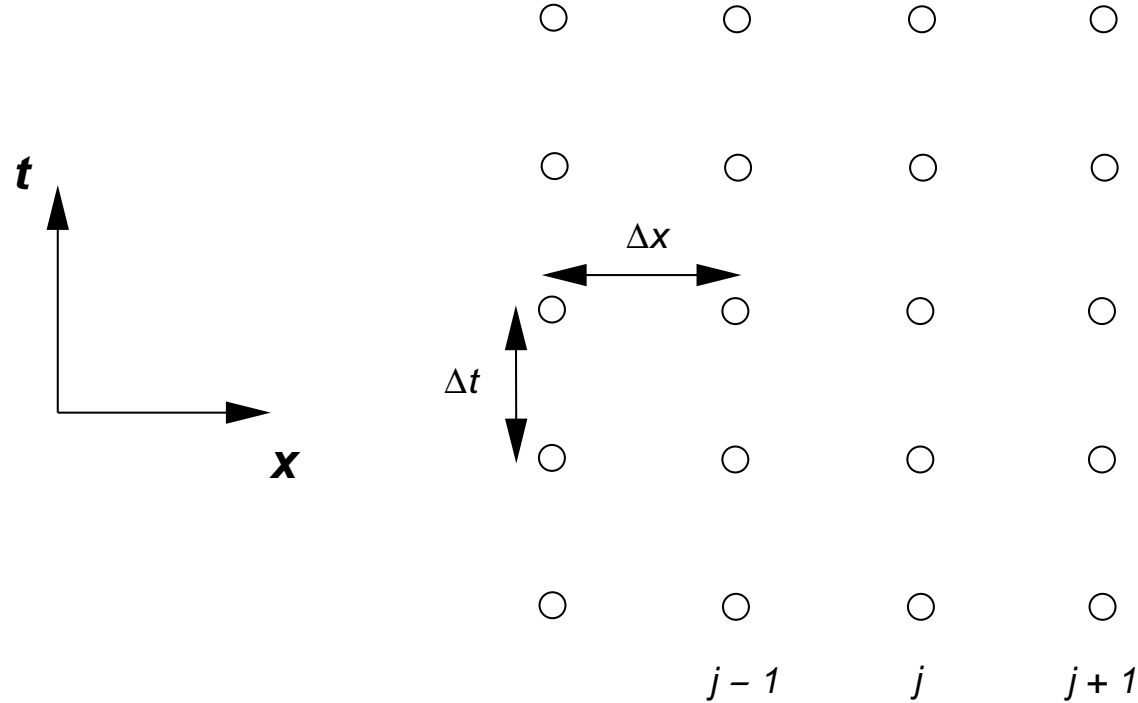
$$\phi_j^n \equiv \phi(n \Delta t, (j - 1) \Delta x)$$

$$\Delta x = (J - 1)^{-1}$$

$$\Delta t = \lambda \Delta x \quad \lambda \equiv \text{“Courant number”}$$

- We note in passing that the quantity λ defined above is often called the Courant number or Courant factor, after the great 20th century mathematician Richard Courant who was a pioneer in the study of finite difference solutions of time dependent PDEs (in particular, in the use of FD techniques to establish existence and uniqueness of such PDEs))

Uniform Grid for 1-D Wave Equation



- When solving wave equations using FDAs, typically keep λ constant when Δx varied.
- FDA will always be characterized by the *single* discretization scale, h .

$$\Delta x \equiv h$$

$$\Delta t \equiv \lambda h$$

FDA for 1-D Wave Equation

- Discretized Interior equation

$$\begin{aligned}(\Delta t)^{-2} \left(\phi_j^{n+1} - 2\phi_j^n + \phi_j^{n-1} \right) &= (\phi_{tt})_j^n + \frac{1}{12} \Delta t^2 (\phi_{tttt})_j^n + O(\Delta t^4) \\ &= (\phi_{tt})_j^n + O(h^2)\end{aligned}$$

$$\begin{aligned}(\Delta x)^{-2} \left(\phi_{j+1}^n - 2\phi_j^n + \phi_{j-1}^n \right) &= (\phi_{xx})_j^n + \frac{1}{12} \Delta x^2 (\phi_{xxxx})_j^n + O(\Delta x^4) \\ &= (\phi_{xx})_j^n + O(h^2)\end{aligned}$$

Putting these two together, get $O(h^2)$ approximation

$$\frac{\phi_j^{n+1} - 2\phi_j^n + \phi_j^{n-1}}{\Delta t^2} = \frac{\phi_{j+1}^n - 2\phi_j^n + \phi_{j-1}^n}{\Delta x^2} \quad j = 2, 3, \dots, J-1 \quad (16)$$

- Scheme such as (16) often called a *three level scheme* since couples *three “time levels”* of data (i.e. unknowns at three distinct, discrete times t^{n-1}, t^n, t^{n+1}).

FDA for 1-D Wave Equation

- Discretized Boundary conditions

$$\phi_1^{n+1} = \phi_J^{n+1} = 0$$

- Discretized Initial conditions

- Need to specify *two* “time levels” of data (effectively $\phi(x, 0)$ and $\phi_t(x, 0)$), i.e. we must specify

$$\phi_j^0 \quad , \quad j = 1, 2, \dots, J$$

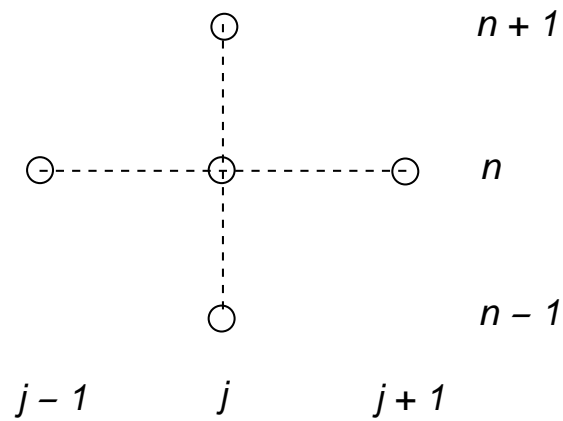
$$\phi_j^1 \quad , \quad j = 1, 2, \dots, J$$

ensuring that the initial values are compatible with the boundary conditions.

- Can solve (16) *explicitly* for ϕ_j^{n+1} :

$$\phi_j^{n+1} = 2\phi_j^n - \phi_j^{n-1} + \lambda^2 \left(\phi_{j+1}^n - 2\phi_j^n + \phi_j^{n-1} \right) \quad (17)$$

Stencil for “Standard” $O(h^2)$ Approximation of 1-D Wave Equation



FDA for 1-D Wave Equation

- Also note that (17) is actually a *linear system* for the unknowns ϕ_j^{n+1} , $j = 1, 2, \dots, J$; in combination with the discrete boundary conditions can write

$$\mathbf{A} \phi^{n+1} = \mathbf{b} \quad (18)$$

where \mathbf{A} is a *diagonal* $J \times J$ matrix and ϕ^{n+1} and \mathbf{b} are vectors of length J .

- Such a difference scheme for an IVP is called an *explicit* scheme.

FDAs: Back to the Basics—Concepts & Definitions

- Will be considering the finite-difference approximation (FDA) of PDEs-0—will generally be interested in the continuum limit, where the *mesh spacing*, or *grid spacing*, usually denoted h , tends to 0.
- Because any specific calculation must necessarily be performed at some specific, *finite* value of h , we will also be (extremely!) interested in the way that our discrete solution varies as a function of h .
- Will *always* view h as the basic “control” parameter of a typical FDA.
- Fundamentally, for sensibly constructed FDAs, we expect the error in the approximation to go to 0, as h goes to 0.

Some Basic Concepts, Definitions and Techniques

- Let

$$Lu = f \tag{54}$$

denote a general *differential* system.

- For simplicity, concreteness, can think of $u = u(x, t)$ as a single function of one space variable and time,
- Discussion applies to cases in more independent variables ($u(x, y, t)$, $u(x, y, z, t)$ \cdots etc.), as well as multiple *dependent* variables ($u = \mathbf{u} = [u_1, u_2, \cdots, u_n]$).
- In (54), L is some differential operator (such as $\partial_{tt} - \partial_{xx}$) in our wave equation example), u is the unknown, and f is some specified function (frequently called a *source* function) of the independent variables.

Some Basic Concepts, Definitions and Techniques

- Here and in the following, will *sometimes* be convenient use notation where a superscript h on a symbol indicates that it is discrete, or associated with the FDA, rather than the continuum.
- With this notation, we will generically denote an FDA of (54) by

$$L^h u^h = f^h \tag{55}$$

where u^h is the discrete solution, f^h is the specified function evaluated on the finite-difference mesh, and L^h is the finite-difference approximation of L .

Residual

- Note that another way of writing our FDA is

$$L^h u^h - f^h = 0 \quad (56)$$

- Often useful to view FDAs in this form for following reasons
 - Have a canonical view of what it means to solve the FDA—“drive the left-hand side to 0”.
 - For iterative approaches to the solution of the FDA (which are common, since it may be too expensive to solve the algebraic equations directly), are naturally lead to the concept of a *residual*.
 - Residual is simply the level of “non-satisfaction” of our FDA (and, indeed, of any algebraic expression).
 - Specifically, if \tilde{u}^h is some approximation to the true solution of the FDA, u^h , then the residual, r^h , associated with \tilde{u}^h is just

$$r^h \equiv L^h \tilde{u}^h - f^h \quad (57)$$

- Leads to the view of a convergent, iterative process as being one which “drives the residual to 0”.

Truncation Error

- *Truncation error*, τ^h , of an FDA is defined by

$$\tau^h \equiv L^h u - f^h \quad (58)$$

where u satisfies the continuum PDE (54).

- Note that the *form* of the truncation error can always be computed (typically using Taylor series) from the finite difference approximation and the differential equations.

Convergence

- Assume FDA is characterized by a *single* discretization scale, h ,
- we say that the approximation *converges* if and only if

$$u^h \rightarrow u \quad \text{as} \quad h \rightarrow 0. \quad (59)$$

- In practice, convergence is clearly our chief concern as numerical analysts, particularly if there is reason to suspect that the solutions of our PDEs are good models for real phenomena.
- Note that this is believed to be the case for many interesting problems in general relativistic astrophysics—the two black hole problem being an excellent example.

Consistency

- Assume FDA with truncation error τ^h is characterized by a single discretization scale, h ,
- Say that the FDA is *consistent* if

$$\tau^h \rightarrow 0 \quad \text{as} \quad h \rightarrow 0. \quad (60)$$

- Consistency is obviously a necessary condition for convergence.

Order of an FDA

- Assume FDA is characterized by a single discretization scale, h
- Say that the FDA is *p-th order accurate* or simply *p-th order* if

$$\lim_{h \rightarrow 0} \tau^h = O(h^p) \quad \text{for some integer } p \quad (61)$$

Solution Error

- Solution error, e^h , associated with an FDA is defined by

$$e^h \equiv u - u^h \tag{62}$$

Relation Between Truncation Error and Solution Error

- Common to tacitly assume that

$$\tau^h = O(h^p) \quad \longrightarrow \quad e^h = O(h^p)$$

- Assumption is often warranted, but is extremely instructive to consider *why* it is warranted and to investigate (following Richardson 1910 (!)) in some detail the *nature* of the solution error.
- Will return to this issue in more detail later.

Error Analysis and Convergence Tests

- Discussion here applies to essentially *any* continuum problem which is solved using FDAs on a *uniform* mesh structure.
- In particular, applies to the treatment of ODEs and elliptic problems
- For such problems convergence is often easier to achieve due to fact that the FDAs are typically intrinsically stable
- Also note that departures from non-uniformity in the mesh do not, in general, complete destroy the picture: however, do tend to distort it in ways that are beyond the scope of these notes.
- **Difficult to overstate importance of convergence studies**

Richardson Ansatz

- **Key idea behind error analysis:** The Richardson ansatz: Appeal to L.F. Richardson's old observation (ansatz), that the solution u^h of any FDA which
 1. Uses a uniform mesh structure with scale parameter h ,
 2. Is completely centeredshould have the following expansion in the limit $h \rightarrow 0$:
 - $u^h(x, t) = u(x, t) + h^2 e_2(x, t) + h^4 e_4(x, t) \dots \quad (72)$
 - Here u is the continuum solution, while e_2, e_4, \dots are (continuum) error functions which do not depend on h .
 - The Richardson expansion (72), is the key expression from which almost all error analysis of FDAs derives.

Convergence Tests

- A simple example of a convergence test, and one commonly used in practice is as follows.
- Compute three distinct FD solutions u^h , u^{2h} , u^{4h} at resolutions h , $2h$ and $4h$ respectively, but using the same initial data (as naturally expressed on the 3 distinct FD meshes).
- Also assume that the finite difference meshes “line up”, i.e. that the $4h$ grid points are a subset of the $2h$ points which are a subset of the h points
- Thus, the $4h$ points constitute a common set of events (x_j, t^n) at which specific grid function values can be directly (i.e. no interpolation required) and meaningfully compared to one another.

Convergence Tests

- From the Richardson *ansatz* (72), expect:

$$\begin{aligned}u^h &= u + h^2 e_2 + h^4 e_4 + \dots \\u^{2h} &= u + (2h)^2 e_2 + (2h)^4 e_4 + \dots \\u^{4h} &= u + (4h)^2 e_2 + (4h)^4 e_4 + \dots\end{aligned}$$

- Then compute a quantity $Q(t)$, which will call a *convergence factor*, as follows:

$$Q(t) \equiv \frac{\|u^{4h} - u^{2h}\|_x}{\|u^{2h} - u^h\|_x} \quad (78)$$

where $\|\cdot\|_x$ is any suitable discrete spatial norm, such as the ℓ_2 norm, $\|\cdot\|_2$:

$$\|u^h\|_2 = \left(J^{-1} \sum_{j=1}^J (u_j^h)^2 \right)^{1/2} \quad (79)$$

- Subtractions in (78) can be taken to involve the sets of mesh points which are common between u^{4h} and u^{2h} , and between u^{2h} and u^h .

Convergence Tests

- Is simple to show that, if the FD scheme is converging, then should find:

$$\lim_{h \rightarrow 0} Q(t) = 4. \quad (80)$$

- In practice, can use additional levels of discretization, $8h$, $16h$, etc. to extend this test to look for “trends” in $Q(t)$ and, in short, to convince oneself (and, with luck, others), that the FDA really *is* converging.
- Additionally, once convergence of an FDA has been established, then point-wise subtraction of any two solutions computed at different resolutions, immediately provides an estimate of the level of error in both.
- For example, if one has u^h and u^{2h} , then, again by the Richardson *ansatz* have

$$u^{2h} - u^h = \left((u + (2h)^2 e_2 + \dots) - (u + h^2 e_2 + \dots) \right) \quad (81)$$

$$= 3h^2 e_2 + O(h^4) \sim 3e^h \sim \frac{3}{4} e^{2h} \quad (82)$$

Richardson Extrapolation

- *Richardson extrapolation*: Richardson's observation (72) also provides the basis for all the techniques of *Richardson extrapolation*
- Solutions computed at different resolutions are linearly combined so as to *eliminate* leading order error terms, providing more accurate solutions.
- As an example, given u^h and u^{2h} which satisfy (72), can take the linear combination

$$\bar{u}^h \equiv \frac{4u^h - u^{2h}}{3} \quad (83)$$

which, by (72), is easily seen to be $O(h^4)$, i.e. *fourth-order* accurate!

$$\begin{aligned} \bar{u}^h &\equiv \frac{4u^h - u^{2h}}{3} = \frac{4(u + h^2e_2 + h^4e_4 + \dots) - (u + 4h^2e_2 + 16h^4e_4 + \dots)}{3} \\ &= -4h^4e_4 + O(h^6) = O(h^4) \end{aligned} \quad (84)$$

Richardson Extrapolation

- When it works, Richardson extrapolation has an almost magical quality about it
- However, generally have to start with fairly accurate (on the order of a few %) solutions in order to see the dramatic improvement in accuracy suggested by (84).
- Still a struggle to achieve that sort of accuracy (i.e. a few %) for *any* computation in many areas of numerical relativity/astrophysics *and* keep the error smooth (which is necessary for Richardson extrapolation to be effective)
- Thus, techniques based on Richardson extrapolation have not had a major impact in this context, although higher-order $O(h^4)$, $O(h^6)$ etc. finite difference methods *are* increasingly common for the vacuum Einstein equations

Independent Residual Evaluation

- Question that often arises in convergence testing: is the following:
“OK, you’ve established that u^h is converging as $h \rightarrow 0$, but how do you know you’re converging to u , the solution of the continuum problem?”
- Here, notion of an independent residual evaluation is very useful.
- Idea is as follows: have continuum PDE

$$Lu - f = 0 \quad (85)$$

and FDA

$$L^h u^h - f^h = 0 \quad (86)$$

- Assume that u^h is apparently converging from, for example, computation of convergence factor (78) that looks like it tends to 4 as h tends to 0.
- However, do not know if we have derived and/or implemented our discrete operator L^h correctly.

Independent Residual Evaluation

- Note that implicit in the “implementation” is the fact that, particularly for multi-dimensional and/or implicit and/or multi-component FDAs, considerable “work” (i.e. analysis and coding) may be involved in setting up and solving the algebraic equations for u^h .
- As a check that solution *is* converging to u , consider a *distinct* (i.e. independent) discretization of the PDE:

$$\tilde{L}^h \tilde{u}^h - f^h = 0 \quad (87)$$

- Only thing needed from this FDA for the purposes of the independent residual test is the new FD operator \tilde{L}^h .
- As with L^h , can expand \tilde{L}^h in powers of the mesh spacing:

$$\tilde{L}^h = L + h^2 E_2 + h^4 E_4 + \dots \quad (88)$$

where E_2, E_4, \dots are higher order (involve higher order derivatives than L) differential operators.

Independent Residual Evaluation

- Now simply apply the new operator \tilde{L}^h to our FDA u^h and investigate what happens as $h \rightarrow 0$.
- If u^h is converging to the continuum solution, u , will have

$$u^h = u + h^2 e_2 + O(h^4) \quad (89)$$

and will compute

$$\tilde{L}^h u^h = (L + h^2 E_2 + O(h^4)) (u + h^2 e_2 + O(h^4)) \quad (90)$$

$$= Lu + h^2 (E_2 u + L e_2) \quad (91)$$

$$= O(h^2) \quad (92)$$

- That is $\tilde{L}^h u^h$ will be a residual-like quantity that converges quadratically as $h \rightarrow 0$.

Stability Analysis

- One of the most frustrating/fascinating features of FD solutions of time dependent problems: discrete solutions often “blow up”—e.g. floating-point overflows are generated at some point in the evolution
- ‘Blow-ups’ can sometimes be caused by legitimate (!) “bugs”—i.e. an incorrect implementation—at other times it is simply the *nature of the FD scheme* which causes problems.
- Are thus lead to consider the *stability* of solutions of difference equations
- Again consider the 1-d wave equation, $u_{tt} = u_{xx}$
- Note that it is a *linear, non-dispersive* wave equation
- Thus the “size” of the solution does *not* change with time:

$$\|u(x, t)\| \sim \|u(x, 0)\|, \quad (95)$$

where $\|\cdot\|$ is an suitable norm, such as the L_2 norm:

$$\|u(x, t)\| \equiv \left(\int_0^1 u(x, t)^2 dx \right)^{1/2}. \quad (96)$$

Stability Analysis

- Will use the property captured by (95) as working definition of stability.
- In particular, if you believe (95) is true for the wave equation, then you believe the wave equation is stable.
- Fundamentally, if FDA approximation *converges*, then expect the same behaviour for the difference solution:

$$\|u_j^n\| \sim \|u_j^0\|. \quad (97)$$

- FD solution constructed by *iterating in time*, generating

$$u_j^0, u_j^1, u_j^2, u_j^3, u_j^4, \dots$$

in succession, using the FD equation

$$u_j^{n+1} = 2u_j^n - u_j^{n-1} + \lambda^2 \left(u_{j+1}^n - 2u_j^n + u_{j-1}^n \right).$$

Stability Analysis

- *Not* guaranteed that (97) holds for all values of $\lambda \equiv \Delta t / \Delta x$.

- For certain λ , have

$$\|u_j^n\| \gg \|u_j^0\|,$$

and for those λ , $\|u^n\|$ *diverges* from u , even (especially!) as $h \rightarrow 0$ —that is, the difference scheme is *unstable*.

- For many wave problems (including all linear problems), given that a FD scheme is *consistent* (i.e. so that $\hat{\tau} \rightarrow 0$ as $h \rightarrow 0$), *stability is the necessary and sufficient condition for convergence* (Lax's theorem).

Heuristic Stability Analysis

- Write general time-dependent FDA in the form

$$\mathbf{u}^{n+1} = \mathbf{G}[\mathbf{u}^n], \quad (98)$$

- \mathbf{G} is some *update operator* (linear in our example problem)
- \mathbf{u} is a column vector containing sufficient unknowns to write the problem in first-order-in-time form.
- Example: introduce new, auxiliary set of unknowns, v_j^n , defined by

$$v_j^n = u_j^{n-1},$$

then can rewrite differenced-wave-equation (16) as

$$u_j^{n+1} = 2u_j^n - v_j^n + \lambda^2 \left(u_{j+1}^n - 2u_j^n + u_{j-1}^n \right), \quad (99)$$

$$v_j^{n+1} = u_j^n, \quad (100)$$

Heuristic Stability Analysis

- Thus with

$$\mathbf{u}^n = [u_1^n, v_1^n, u_2^n, v_2^n, \dots, u_J^n, v_J^n],$$

(for example), (99-100) is of the form (98).

- Equation (98) provides compact way of describing the FDA solution.
- Given initial data, \mathbf{u}^0 , solution after n time-steps is

$$\mathbf{u}^n = \mathbf{G}^n \mathbf{u}^0, \quad (101)$$

where \mathbf{G}^n is the n -th power of the matrix \mathbf{G} .

- Assume that \mathbf{G} has a complete set of orthonormal eigenvectors

$$\mathbf{e}_k, \quad k = 1, 2, \dots, J,$$

and corresponding eigenvalues

$$\mu_k, \quad k = 1, 2, \dots, J,$$

Heuristic Stability Analysis

- Thus have

$$\mathbf{G} \mathbf{e}_k = \mu_k \mathbf{e}_k, \quad k = 1, 2, \dots, J.$$

- Can then write initial data as (spectral decomposition):

$$\mathbf{u}^0 = \sum_{k=1}^J c_k^0 \mathbf{e}_k,$$

where the c_k^0 are coefficients.

- Using (101), solution at time-step n is

$$\mathbf{u}^n = \mathbf{G}^n \left(\sum_{k=1}^J c_k^0 \mathbf{e}_k \right) \quad (102)$$

$$= \sum_{k=1}^J c_k^0 (\mu_k)^n \mathbf{e}_k. \quad (103)$$

Heuristic Stability Analysis

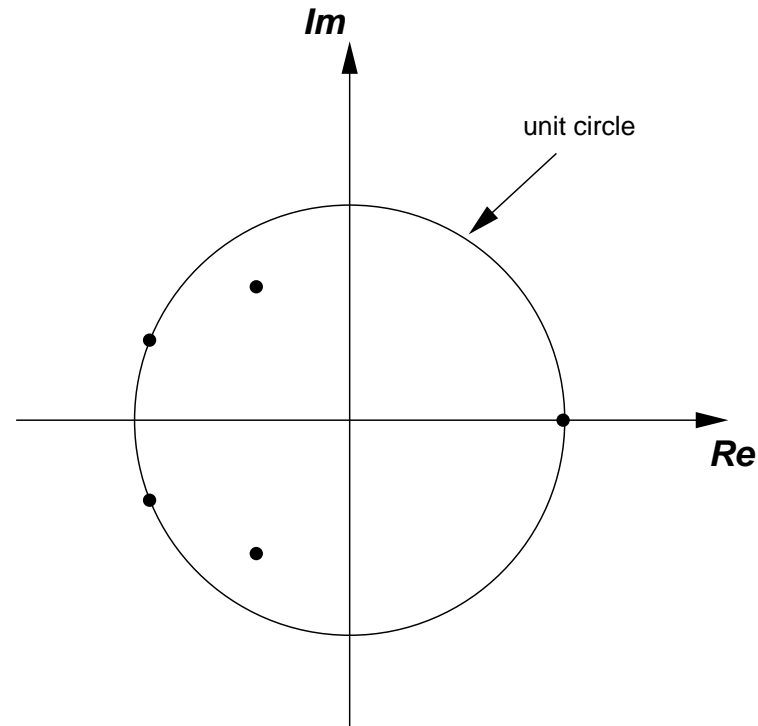
- If difference scheme is to be stable, must have

$$|\mu_k| \leq 1 \quad k = 1, 2, \dots, J \quad (104)$$

(Note: μ_k will be complex in general, so $|\mu|$ denotes the complex modulus, $|\mu| \equiv \sqrt{\mu\mu^*}$).

- Geometric interpretation: eigenvalues of the update matrix must lie on or within the unit circle

Heuristic Stability Analysis



- Schematic illustration of location in complex plane of eigenvalues of update matrix \mathbf{G} .
- In this case, all eigenvalues (dots) lie on or within the unit circle, indicating that the corresponding finite difference scheme is stable.

Von-Neumann (Fourier) Stability Analysis (Summary)

- Von-Neumann (VN) stability analysis based on the ideas sketched above
- Assumes that difference equation is linear with constant coefficients, periodic boundary conditions boundary conditions are periodic
- Can then use Fourier analysis: difference operators in real-space variable $x \longrightarrow$ algebraic operations in Fourier-space variable k
- VN applied to wave-equation example shows that must have

$$\lambda \equiv \frac{\Delta t}{\Delta x} \leq 1,$$

for stability of scheme (16).

- Condition is often called the CFL condition—after Courant, Friedrichs and Lewy who derived it in 1928
- This type of instability has “physical” interpretation, often summarized by the statement *the numerical domain of dependence of an explicit difference scheme must contain the physical domain of dependence.*

1-D Wave Equation: 1st Order Form

- Let us again consider the 1-D wave equation, solved on the spatial domain $0 \leq x \leq 1$, and where we will delay the specification of the boundary conditions for the time being
- We have

$$\phi_{tt} = \phi_{xx}, \quad 0 \leq x \leq 1, \quad t \geq 0 \quad (19)$$

$$\phi(0, x) = \phi_0(x) \quad (20)$$

$$\phi_t(0, x) = \Pi_0(x) \quad (21)$$

- We rewrite (19) in a form that involves only first time derivatives by defining the following auxiliary variables

$$\Phi(t, x) \equiv \phi_x \quad (22)$$

$$\Pi(t, x) \equiv \phi_t \quad (23)$$

1-D Wave Equation: 1st Order Form

- Using the commutativity of (mixed) partial derivatives, it is easy to show that (19) is equivalent to the following system

$$\Phi_t = \Pi_x \quad (24)$$

$$\Pi_t = \Phi_x \quad (25)$$

- The initial conditions are then given by

$$\Phi(0, x) = \frac{d}{dx}\phi_0(x) \quad (26)$$

$$\Pi(0, x) = \Pi_0(x) \quad (27)$$

- We also note that if we are not concerned with actually computing values of the scalar field, $\phi(t, x)$ itself (and in this treatment we will *not* be), then we can equally well replace (26) with

$$\Phi(0, x) = \Phi_0(x) \quad (28)$$

i.e. we can specify the initial values of $\Phi \equiv \phi_x$ *directly*

1-D Wave Equation: 1st Order Form

- We now return to the issue of boundary conditions: we wish to illustrate a type of boundary condition which is often imposed when a (pure) initial-value problem for a hyperbolic system has been converted into an initial-boundary-value problem by truncation of the solution domain to some finite extent
- Thus, although we will solve the wave equation on the spatial domain $0 \leq x \leq 1$, we want the solution to approximate the one that we would get if we were able to solve on the unbounded domain $-\infty < x < \infty$
- We assume that the initial conditions represent some set of disturbances which are localized in space, well away from the boundaries $x = 0$ and $x = 1$, and that the subsequent dynamics describes the propagation of these disturbances in and away from the interval in which they are initially localized
- We recall that the general solution of the wave equation can be written in the form

$$\phi(t, x) \sim \ell(x + t) + r(x - t) \quad (29)$$

where ℓ and r are the left- and right-moving parts of the solution, respectively

1-D Wave Equation: 1st Order Form

- We further observe that it follows from (29) and the definitions of Φ and Π that $\Phi \equiv \phi_x$ and $\Pi \equiv \phi_t$ can also be written as a linear combination of right- and left-moving pieces
- The boundary condition we now wish to employ is often called a radiation condition, or Sommerfeld condition, and is equivalent to the demand that there be no incoming radiation (disturbances) at the boundaries of the solution domain
- This means that at $x = 0$ we must have only left-moving signals, so that $\Phi(t, x) \sim \Phi(x + t)$ and $\Pi(t, x) \sim \Pi(x + t)$, or

$$\Phi_t(t, 0) = \Phi_x(t, 0) \quad (30)$$

$$\Pi_t(t, 0) = \Pi_x(t, 0) \quad (31)$$

- Similarly, at $x = 1$ we require only left-moving waves, so that $\Phi(t, x) \sim \Phi(x - t)$ and $\Pi(t, x) \sim \Pi(x - t)$, or

$$\Phi_t(t, 1) = -\Phi_x(t, 1) \quad (32)$$

$$\Pi_t(t, 1) = -\Pi_x(t, 1) \quad (33)$$

1-D Wave Equation: Crank-Nicholson Scheme

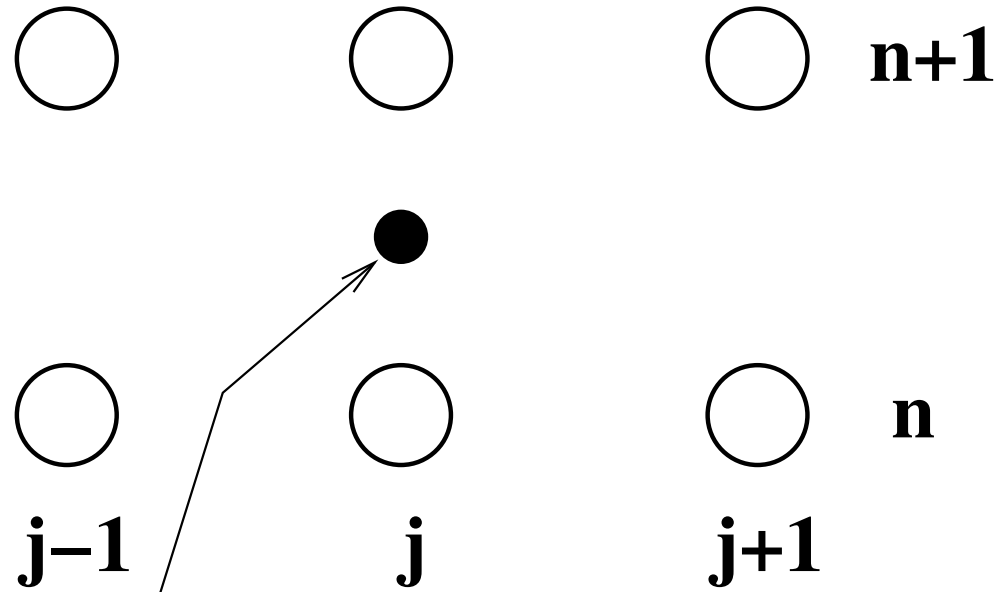
- We now discuss the Crank-Nicholson discretization scheme for the 1-D wave equation as written in the first order form defined above: variations on this theme will be used extensively in this week's lectures and tutorial sessions
- We adopt the same uniform grid structure (in space and time) as previously, but now use the stencil illustrated on the next page for our PDEs

$$\Phi_t = \Pi_x$$

$$\Pi_t = \Phi_x$$

- In our description of the Crank Nicholson FDA we will also introduce the notion of finite difference operators, which provide a compact way of denoting many FDAs, and which play a central role in the special purpose programming language, RNPL, that we will use in the tutorial sessions

Stencil for $O(h^2)$ Crank-Nicholson Approximation of 1-D Wave Equation



Scheme is centred at $t^{n+1/2}$, x_j

1-D Wave Equation: Crank-Nicholson Scheme

- To illustrate the scheme, it will suffice to consider one of the two first-order PDEs that together constitute the wave equation: for specificity we focus on

$$\Phi_t = \Pi_x \quad (34)$$

- The time derivative of Φ is approximated using

$$\begin{aligned} \Delta t^{-1} \left(\Phi_j^{n+1} - \Phi_j^n \right) &= (\Phi_t)_j^{n+\frac{1}{2}} + \frac{1}{24} \Delta t^2 (\Phi_{ttt})_j^{n+\frac{1}{2}} + O(\Delta t^4) \quad (35) \\ &= (\Phi_t)_j^{n+\frac{1}{2}} + O(\Delta t^2) \end{aligned}$$

- To approximate Π_x , we write the usual $O(h^2)$ centred approximation for the first derivative in operator form as

$$D_x \Pi_j^n \equiv (2 \Delta x)^{-1} \left(\Pi_{j+1}^n - \Pi_{j-1}^n \right) \quad (36)$$

$$D_x = \partial_x + \frac{1}{6} \Delta x^2 \partial_{xxx} + O(\Delta x^4) \quad (37)$$

1-D Wave Equation: Crank-Nicholson Scheme

- We further introduce the (forward) time-averaging operator, μ_t :

$$\mu_t u_j^n \equiv \frac{1}{2} \left(u_j^{n+1} + u_j^n \right) = u_j^{n+\frac{1}{2}} + \frac{1}{8} \Delta t^2 (u_{tt})_j^{n+\frac{1}{2}} + O(\Delta t^4) \quad (38)$$

$$\mu_t = \left[I + \frac{1}{8} \Delta t^2 \partial_{tt} + O(\Delta t^4) \right]_{t=t^{n+1/2}} \quad (39)$$

where I is the identity operator.

- Assuming that $\Delta t = O(\Delta x) = O(h)$, it is easy to show (**exercise**) that

$$\mu_t \left[D_x \Pi_j^n \right] = (\Pi_x)_j^{n+\frac{1}{2}} + O(h^2)$$

- Putting above results together, we get the ($O(h^2)$) Crank-Nicholson approximation of $\Phi_t = \Pi_x$

$$\frac{\Phi_j^{n+1} - \Phi_j^n}{\Delta t} = \mu_t \left[D_x \Pi_j^n \right] \quad (40)$$

1-D Wave Equation: Crank-Nicholson Scheme

- Written out in full, this is

$$\frac{\Phi_j^{n+1} - \Phi_j^n}{\Delta t} = \frac{1}{2} \left[\frac{\Pi_{j+1}^{n+1} - \Pi_{j-1}^{n+1}}{2 \Delta x} + \frac{\Pi_{j+1}^n - \Pi_{j-1}^n}{2 \Delta x} \right] \quad (41)$$

- Note that the Crank-Nicholson scheme immediately generalizes to any equation that can be written in the form

$$u_t = L[u] \quad (42)$$

where L is some spatial operator. A Crank-Nicholson FDA of (42) is

$$\frac{u_j^{n+1} - u_j^n}{\Delta t} = \frac{1}{2} (L^h [u^{n+1}] + L^h [u^n]) \quad (43)$$

where L^h is some discretization of L , not necessarily second order

- Also observe that Crank-Nicholson scheme is a two-level method (couples unknowns at two discrete time steps)

1-D Wave Equation: Crank-Nicholson Scheme

- The difference equations (40) can be applied at grid points labelled by $j = 2, 3, \dots, J - 1$ (the interior points)
- For $j = 1$ and $j = J$ we use discretized versions of the radiation (Sommerfeld) boundary conditions

$$\Phi_t(t, 0) = \Phi_x(t, 0) \quad (44)$$

$$\Phi_t(t, 1) = -\Phi_x(t, 1) \quad (45)$$

- The time derivatives are approximated as previously, and for the space derivatives we use second order, forward and backward (“off-centred”) difference approximations defined by

$$D_x^F \Phi_j^n \equiv (2 \Delta x)^{-1} \left(-3\Phi_j^n + 4\Phi_{j+1}^n - \Phi_{j+2}^n \right) \quad (46)$$

$$D_x^F = \partial_x + O(\Delta x^2) \quad \text{exercise} \quad (47)$$

$$D_x^B \Phi_j^n \equiv (2 \Delta x)^{-1} \left(3\Phi_j^n - 4\Phi_{j-1}^n + \Phi_{j-2}^n \right) \quad (48)$$

$$D_x^B = \partial_x + O(\Delta x^2) \quad \text{exercise} \quad (49)$$

1-D Wave Equation: Crank-Nicholson Scheme

- Employing the time-averaging operator, μ_t , defined previously, the FDAs for the outgoing-radiation boundary conditions are

$$\frac{\Phi_j^{n+1} - \Phi_j^n}{\Delta t} = \mu_t \left[D_x^F \Phi_j^n \right] \quad j = 1 \quad (50)$$

$$\frac{\Phi_j^{n+1} - \Phi_j^n}{\Delta t} = -\mu_t \left[D_x^B \Phi_j^n \right] \quad j = J \quad (51)$$

- Finally, in our RNPL implementation of this scheme, we will set initial data of the form

$$\Phi_j^0 = A \exp \left[-((x - x_0) / \delta)^2 \right] \quad (52)$$

$$\Pi_j^0 = \sigma \Phi_j^0 \quad (53)$$

where $\sigma = -1, 0, 1$ will generate purely left-moving, left-moving/right-moving (time symmetric) or purely right-moving data, respectively, and where A , x_0 and δ are adjustable parameters of the gaussian pulse shape