Laurette TUCKERMAN

laurette@pmmh.espci.fr

Numerical Methods for Differential Equations in Physics

Discretization in Two or Three Dimensions

Classification of second order linear partial differential equations

$$au_{xx} + bu_{xy} + c_{yy} + du_x + eu_y + fu = g$$

Depends on coefficients a, b, c. Compare to quadratic functions:

$$ax^2 + bxy + cy^2 + dx + ey = g$$

Ellipse: $b^2 - 4ac < 0$

Examples: $x^2 + y^2 = g$

Poisson $\left(\partial_{xx} + \partial_{yy}\right) u = g$

Laplace $\left(\partial_{xx} + \partial_{yy}\right) u = 0$

Helmholtz $\left(\partial_{xx} + \partial_{yy} - k^2 I\right) u = 0$

Parabola: $b^2 - 4ac = 0$

Examples: $x^2 + y = g$

Heat $\partial_t u = \partial_{xx} u$

Hyperbola: $b^2 - 4ac > 0$

Examples: $x^2 - y^2 = g$

Wave $\partial_{tt} u = \partial_{xx} u$

Elliptic Partial Differential Equations: Poisson or Laplace equation

$$b^2 - 4ac < 0$$

$$\Delta u = u_{xx} + u_{yy} = g$$

Periodic boundary conditions in x and y: Fourier-Fourier

$$u(x,y) = \sum_{k,m} \hat{u}_{k,m} e^{ikx} e^{imy} \qquad g(x,y) = \sum_{k,m} \hat{g}_{k,m} e^{ikx} e^{imy}$$

$$\Delta u = \sum_{k,m} (-k^2 - m^2) \hat{u}_{k,m} e^{ikx} e^{imy} = \sum_{k,m} \hat{g}_{k,m} e^{ikx} e^{imy}$$

$$\hat{u}_{k,m} = \frac{-\hat{g}_{k,m}}{k^2 + m^2}$$

Have used interval $[0, 2\pi)$ for simplicity. More generally,

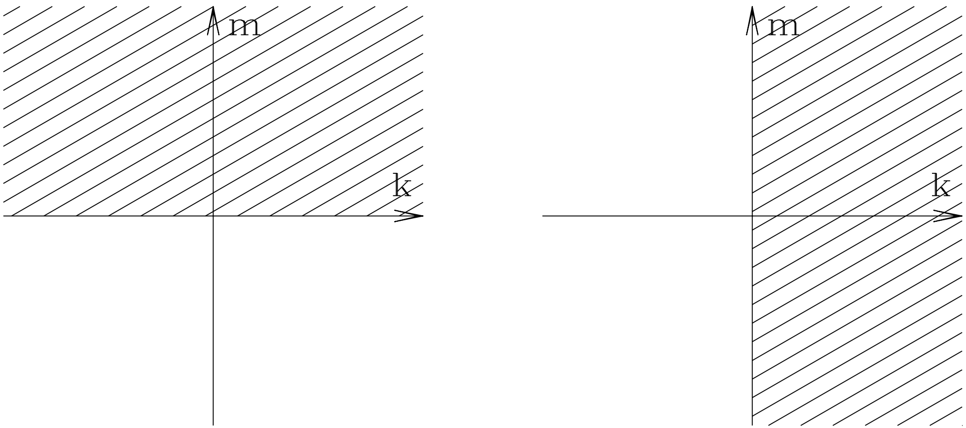
domain is $[0, L_x) \times [0, L_y)$ and basis functions are $e^{ikx2\pi/L_x} e^{imy2\pi/L_y}$

For $u(x)$ real, $\hat{u}_k$ is conjugate symmetric: $\hat{u}_{-k} = \hat{u}_k^*$ so that

$$u(x) \sim \hat{u}_k e^{ikx} + \hat{u}_{-k} e^{-ikx} = \hat{u}_k e^{ikx} + \left(\hat{u}_k e^{ikx}\right)^* = 2\mathcal{R}e\left(\hat{u}_k e^{ikx}\right)$$

is real.

For $u(x, y)$ real, $\hat{u}_{-k,-m} = \hat{u}_{k,m}^*$ so need half of (k, m) plane.



Multidimensional Fourier Transform:

Transform in x: N_y independent transforms, each takes time $O(N_x \log N_x)$.
Transform in y: N_x independent transforms, each takes time $O(N_y \log N_y)$.
Total: $N_y N_x \log N_x + N_x N_y \log N_y = N_x N_y \log(N_x N_y)$.

Three dimensions (x, y, z):
Total: $N_x N_y N_z (\log N_x + \log N_y + \log N_z) = N_x N_y N_z \log(N_x N_y N_z)$

Even without FFT (SFT= Slow Fourier Transform), multidimensional Fourier transform would be fast because the different dimensions are decoupled:
$N_x N_y N_z (N_x + N_y + N_z)$
Decoupling also applies in other non-Fourier contexts.

Fourier transform in x is action with matrix $F^x_{k_x,k'_x} \delta_{k_y,k'_y} \delta_{k_z,k'_z}$
Fourier transform in y is action with matrix $F^y_{k_y,k'_y} \delta_{k_x,k'_x} \delta_{k_z,k'_z}$
Fourier transform in z is action with matrix $F^z_{k_z,k'_z} \delta_{k_y,k'_y} \delta_{k_x,k'_x}$

Periodic boundary conditions in x, Dirichlet boundary conditions in y: Fourier-Finite Differences

$$u(x,y) = \sum_k \hat{u}_k(y) e^{ikx} \qquad\qquad g(x,y) = \sum_k \hat{g}_k(y) e^{ikx}$$

$$\begin{aligned}\Delta u &= \sum_k \left(-k^2 \hat{u}_k(y) + \frac{\hat{u}_k(y+\Delta y) - 2\hat{u}_k(y) + \hat{u}_k(y-\Delta y)}{(\Delta y)^2} \right) e^{ikx} \\ &= \sum_k \frac{\hat{u}_k(y+\Delta y) - \left(2 + k^2 (\Delta y)^2\right) \hat{u}_k(y) + \hat{u}_k(y-\Delta y)}{(\Delta y)^2} e^{ikx}\end{aligned}$$

$$\frac{1}{(\Delta y)^2} \begin{pmatrix} -(2+k^2\Delta y) & 1 & & & \\ 1 & -(2+k^2\Delta y) & 1 & & \\ & 1 & -(2+k^2\Delta y) & 1 & \\ & & & \ddots & \\ & & & 1 & -(2+k^2\Delta y) \end{pmatrix} \begin{pmatrix} \hat{u}_k(y_1) \\ \hat{u}_k(y_2) \\ \hat{u}_k(y_3) \\ \vdots \\ \hat{u}_k(y_N) \end{pmatrix}$$

Boundary conditions needed, e.g.

$$\frac{1}{(\Delta y)^2}\begin{pmatrix} 1 & 0 & 0 & \dots & 0 \\ 1 & -(2+k^2\Delta y) & 1 & & \\ & 1 & -(2+k^2\Delta y) & 1 & \\ & & & \ddots & \\ 0 & 0 & 0 & \dots & 1 \end{pmatrix}\begin{pmatrix} \hat{u}_k(y_1) \\ \hat{u}_k(y_2) \\ \hat{u}_k(y_3) \\ \vdots \\ \hat{u}_k(y_N) \end{pmatrix} = \begin{pmatrix} \gamma_- \\ \hat{g}_k(y_2) \\ \hat{g}_k(y_3) \\ \vdots \\ \gamma_+ \end{pmatrix}$$

Elliptic equations need boundary conditions (Dirichlet, Neumann or periodic) along all boundaries of domain

How should a system like this be solved?

LU decomposition $\Longleftrightarrow$ Gaussian elimination

$$\begin{pmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{pmatrix} = \begin{pmatrix} 1 & & & \\ \ell_{21} & 1 & & \\ \ell_{31} & \ell_{32} & 1 & \\ \ell_{41} & \ell_{42} & \ell_{43} & 1 \end{pmatrix} \begin{pmatrix} u_{11} & u_{12} & u_{13} & u_{14} \\ & u_{22} & u_{23} & u_{24} \\ & & u_{33} & u_{34} \\ & & & u_{44} \end{pmatrix}$$

$$\begin{pmatrix} 1 & & & \\ \ell_{21} & 1 & & \\ \ell_{31} & \ell_{32} & 1 & \\ \ell_{41} & \ell_{42} & \ell_{43} & 1 \end{pmatrix} \underbrace{\left[\begin{pmatrix} u_{11} & u_{12} & u_{13} & u_{14} \\ & u_{22} & u_{23} & u_{24} \\ & & u_{33} & u_{34} \\ & & & u_{44} \end{pmatrix} \begin{pmatrix} f_1 \\ f_2 \\ f_3 \\ f_4 \end{pmatrix} \right]}_{h} = \begin{pmatrix} g_1 \\ g_2 \\ g_3 \\ g_4 \end{pmatrix}$$

$$\begin{pmatrix} u_{11} & u_{12} & u_{13} & u_{14} \\ & u_{22} & u_{23} & u_{24} \\ & & u_{33} & u_{34} \\ & & & u_{44} \end{pmatrix} \begin{pmatrix} f_1 \\ f_2 \\ f_3 \\ f_4 \end{pmatrix} = \begin{pmatrix} h_1 \\ h_2 \\ h_3 \\ h_4 \end{pmatrix}$$

Find $f_4 = h_4/u_{44}$**, then** $f_3 = (h_3 - u_{34} f_4)/u_{33}, \ldots$

LU decomposition preserves bandedness

$$\begin{pmatrix} a_{11} & a_{12} & & \\ a_{21} & a_{22} & a_{23} & \\ & a_{32} & a_{33} & a_{34} \\ & & a_{43} & a_{44} \end{pmatrix} = \begin{pmatrix} 1 & & & \\ \ell_{21} & 1 & & \\ & \ell_{32} & 1 & \\ & & \ell_{43} & 1 \end{pmatrix} \begin{pmatrix} u_{11} & u_{12} & & \\ & u_{22} & u_{23} & \\ & & u_{33} & u_{34} \\ & & & u_{44} \end{pmatrix}$$

For $N \times N$ matrix, operation count is $O(N^3)$ for LU decomposition and $O(N^2)$ for backsolve.

For $N \times N$ matrix with J diagonal bands, operation count is $O(J^2 N)$ for LU decomposition and $O(JN)$ for backsolve.

LU decomposition is done once, backsolve done for each right-hand-side.

Periodic boundary conditions in x, Dirichlet boundary conditions in y: Fourier-Chebyshev

$$u(x,y) = \sum_{k,n} u_{k,n} e^{ikx} T_n(y) \qquad g(x,y) = \sum_{k,n} g_{k,n} e^{ikx} T_n(y)$$

$$\begin{aligned}
\Delta u &= \sum_{k,n} -k^2 u_{k,n} e^{ikx} T_n(y) + \sum_{k,n} u_{k,n} e^{ikx} T_n''(y) \\
&= \sum_{k,n} -k^2 u_{k,n} e^{ikx} T_n(y) + \sum_{k,n} u_{k,n} e^{ikx} \sum_m R_{m,n} T_m(y) \\
&= \sum_{k,n} -k^2 u_{k,n} e^{ikx} T_n(y) + \sum_{k,m} \left(\sum_n R_{m,n} u_{k,n} \right) e^{ikx} T_m(y) \\
&= \sum_{k,n} -k^2 u_{k,n} e^{ikx} T_n(y) + \sum_{k,m} (R u_k)_m e^{ikx} T_m(y) \\
&= \sum_{k,n} -k^2 u_{k,n} e^{ikx} T_n(y) + \sum_{k,n} (R u_k)_n e^{ikx} T_n(y) \\
&= \sum_{k,n} \left(-k^2 u_{k,n} + (R u_k)_n \right) e^{ikx} T_n(y)
\end{aligned}$$

where $(Ru_k)_m \equiv \sum_m R_{m,n} u_{k,n}$

$$\boldsymbol{\Delta u(x,y) = \sum_{k,n} \left(\sum_{k',n'} \Delta_{k,n,k',n'} u_{k',n'} \right) e^{ikx} T_n(y)}$$

where $$\boldsymbol{\Delta_{k,n,k',n'} = -k^2 \delta_{k,k'} \delta_{n,n'} + R_{n,n'} \delta_{k,k'}}$$

Operators in x commute with operators in y.

Odd and even Chebyshev polynomials are decoupled.

Recall: second-derivative Chebyshev matrix R is upper triangular tridiagonal matrix B is such that BR is tridiagonal.

$$\boldsymbol{B\Delta u(x,y) = \sum_{k,n,n'} \left(-k^2 B_{n,n'} + (BR)_{n,n'} \right) u_{k',n'} e^{ikx} T_n(y)}$$

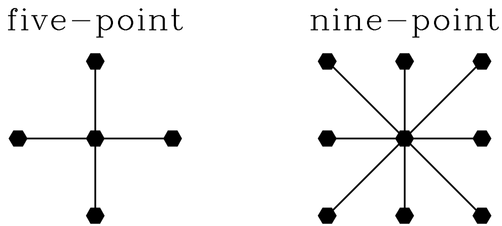
Boundary conditions needed

$$
\begin{pmatrix}
1 & 1 & 1 & 1 & 1 \\
(BR - k^2B)_{2,0} & (BR - k^2B)_{2,2} & (BR - k^2B)_{2,4} & & \\
 & (BR - k^2B)_{4,2} & (BR - k^2B)_{4,4} & (BR - k^2B)_{4,6} & \\
 & & (BR - k^2B)_{6,4} & (BR - k^2B)_{6,6} & (BR - k^2B)_{6,8}
\end{pmatrix}
$$

LU preserves bandedness and full rows at bottom or columns on right.
UL preserves bandedness and full rows at top or columns on left.

$$
\begin{bmatrix}
\times & \times & \times & \times & \times & \times & \times & \times & \times & \times & \times \\
\times & \times & \times & & & & & & & & \times \\
 & \times & \times & \times & & & & & & & \times \\
 & & \times & \times & \times & & & & & & \times \\
 & & & \times & \times & \times & & & & & \times \\
 & & & & \times & \times & \times & & & & \times \\
 & & & & & \times & \times & \times & & & \times \\
 & & & & & & \times & \times & \times & & \times \\
 & & & & & & & \times & \times & \times & \times \\
 & & & & & & & & \times & \times & \times \\
 & & & & & & & & & \times & \times
\end{bmatrix}
=
\begin{bmatrix}
\times & \times & \times & \times & \times & \times & \times & \times & \times & \times & \times \\
 & \times & \times & & & & & & & & \times \\
 & & \times & \times & & & & & & & \times \\
 & & & \times & \times & & & & & & \times \\
 & & & & \times & \times & & & & & \times \\
 & & & & & \times & \times & & & & \times \\
 & & & & & & \times & \times & & & \times \\
 & & & & & & & \times & \times & & \times \\
 & & & & & & & & \times & \times & \times \\
 & & & & & & & & & \times & \times \\
 & & & & & & & & & & \times
\end{bmatrix}
\begin{bmatrix}
1 & & & & & & & & & & \\
\times & 1 & & & & & & & & & \\
 & \times & 1 & & & & & & & & \\
 & & \times & 1 & & & & & & & \\
 & & & \times & 1 & & & & & & \\
 & & & & \times & 1 & & & & & \\
 & & & & & \times & 1 & & & & \\
 & & & & & & \times & 1 & & & \\
 & & & & & & & \times & 1 & & \\
 & & & & & & & & \times & 1 & \\
 & & & & & & & & & \times & 1
\end{bmatrix}
$$

Dirichlet BCs in x and y: Finite differences/Finite differences



Five-point stencil: $$\frac{d^2u}{dx^2}(x_j, y_k) + \frac{d^2u}{dy^2}(x_j, y_k)$$

$$\approx \frac{1}{h^2}\left(u(x_{j+1}, y_k) - 2u(x_j, y_k) + u(x_{j-1}, y_k)\right)$$

$$+ \frac{1}{h^2}\left(u(x_j, y_{k+1}) - 2u(x_j, y_k) + u(x_j, y_{k-1})\right)$$

Error is $$\frac{h^2}{24}(\partial_x^4 + \partial_y^4)u + \dots$$

This error is not isotropic, unlike the Laplacian itself.

We show that the continuous Laplacian IS isotropic:

Rotate $(x,y) \to (x',y')$: $\quad x' = \alpha x + \beta y, \quad y' = -\beta x + \alpha y, \quad \alpha^2 + \beta^2 = 1$

$$\frac{\partial}{\partial x} = \frac{\partial x'}{\partial x}\frac{\partial}{\partial x'} + \frac{\partial y'}{\partial x}\frac{\partial}{\partial y'} = \alpha\frac{\partial}{\partial x'} - \beta\frac{\partial}{\partial y'}$$

$$\frac{\partial^2}{\partial x^2} = \left(\alpha\frac{\partial}{\partial x'} - \beta\frac{\partial}{\partial y'}\right)^2$$

$$= \alpha^2\frac{\partial^2}{\partial x'^2} - 2\alpha\beta\frac{\partial^2}{\partial x'\partial y'} + \beta^2\frac{\partial^2}{\partial y'^2}$$

$$\frac{\partial}{\partial y} = \frac{\partial x'}{\partial y}\frac{\partial}{\partial x'} + \frac{\partial y'}{\partial y}\frac{\partial}{\partial y'} = \beta\frac{\partial}{\partial x'} + \alpha\frac{\partial}{\partial y'}$$

$$\frac{\partial^2}{\partial y^2} = \left(\beta\frac{\partial}{\partial x'} + \alpha\frac{\partial}{\partial y'}\right)^2$$

$$= \beta^2\frac{\partial^2}{\partial x'^2} + 2\alpha\beta\frac{\partial^2}{\partial x'\partial y'} + \alpha^2\frac{\partial^2}{\partial y'^2}$$

$$\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} = \frac{\partial^2}{\partial x'^2} + \frac{\partial^2}{\partial y'^2}$$

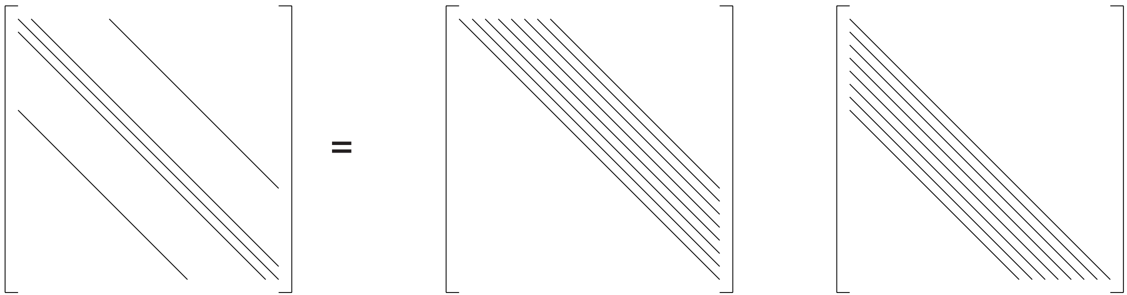
Nine-point stencil is isotropic:

$$\frac{d^2u}{dx^2}(x_j, y_k) + \frac{d^2u}{dy^2}(x_j, y_k) \approx$$

$$\frac{1}{6h^2}\left[4\left(u(x_{j+1}, y_k)u(x_{j-1}, y_k) + u(x_j, y_{k+1}) + u(x_j, y_{k-1})\right)\right.$$
$$+ u(x_{j+1}, y_{k+1}) + u(x_{j-1}, y_{k-1}) + u(x_{j+1}, y_{k-1}) + u(x_{j-1}, y_{k+1})$$
$$\left. -20u(x_j, y_k)\right]$$

Error: $\frac{h^2}{12}\left(\partial_x^2 + \partial_y^2\right)^2 = \frac{h^2}{12}\Delta^2$

Two-dimensional finite-difference Laplacian:

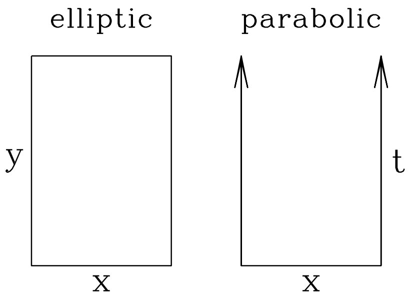


Fill-in until maximal bandwidth: bandedness not preserved
Alternative way of inverting: diagonalize in one or both directions.
Recall that operators in x commute with operators in y
so they are simultaneously diagonalizable.

$$\begin{aligned}\Delta(n,m,n',m') &= D^{xx}_{n,n'}\delta_{m,m'} + D^{yy}_{m,m'}\delta_{n,n'} \\ &= V^{xx}\Lambda^{xx}\left(V^{xx}\right)^{-1}\delta_{m,m'} + V^{yy}\Lambda^{yy}\left(V^{yy}\right)^{-1}\delta_{n,n'}\end{aligned}$$

Transform to x and y eigenspace $O(N_x^2 N_y + N_y^2 N_x)$
Invert Laplacian in eigenspace: $O(N_x N_y)$
Inverse transform back from x and y eigenspace $O(N_x^2 N_y + N_y^2 N_x)$
Total: $O(N_x N_y (N_x + N_y))$.

What about other kinds of PDEs?
Parabolic, example of heat equation: $\partial_t u = \partial_{xx} u$



Need:
boundary conditions on boundary in x
(possibly also y, z **in higher dimensions)**
initial condition at $t = 0$

Spatial discretization: Finite differences, Fourier, Chebyshev

Temporal discretization: implicit or explicit multistep or Runge-Kutta

Implicit Euler:

$$U(t + \Delta t) = U(t) + \Delta t D^{xx} U(t + \Delta t)$$
$$(I - \Delta t D^{xx}) U(t + \Delta t) = U(t)$$

Helmholtz equation – like Poisson equation
Same questions of discretization and solution of linear systems

Example of Navier-Stokes equations

$$\partial_t \mathrm{U} = -(\mathrm{U} \cdot \nabla)\mathrm{U} - \nabla P + Re^{-1}\Delta \mathrm{U}$$

$$\nabla \cdot \mathrm{U} = 0$$

$$\mathrm{U} = \mathrm{b} \text{ at boundaries}$$

Four equations for four fields: U, V, W, P.
Derive pressure Poisson equation using $\nabla \cdot \mathrm{U} = 0$

$$\underbrace{\nabla \cdot \partial_t \mathrm{U}}_{0} = -\nabla \cdot (\mathrm{U} \cdot \nabla)\mathrm{U} - \nabla \cdot \nabla P + Re^{-1} \underbrace{\nabla \cdot \Delta \mathrm{U}}_{0}$$

$$\boxed{\Delta P = -\nabla \cdot (\mathrm{U} \cdot \nabla)\mathrm{U}}$$

Use implicit Euler for viscous term, explicit Euler for other terms

$$\mathrm{U}(t + \Delta t) = U(t) - \Delta t \nabla P(t) - \Delta t (\mathrm{U} \cdot \nabla)\mathrm{U}(t) + Re^{-1}\Delta t \Delta \mathrm{U}(t + \Delta t)$$

$$\boxed{(I - Re^{-1}\Delta t \Delta)\mathrm{U}(t + \Delta t) = U(t) - \Delta t \nabla P(t) - \Delta t (\mathrm{U} \cdot \nabla)\mathrm{U}(t)}$$

Need to solve Poisson and Helmholtz equations at each timestep.

$$\partial_t \mathrm{U} = -(\mathrm{U}\cdot\nabla)\mathrm{U} - \nabla P + Re^{-1}\Delta\mathrm{U}$$

Timestepping the advective term explicitly:

$$U(t+\Delta t) = U(t) - \Delta t(\mathrm{U}(t)\cdot\nabla)\mathrm{U}(t)$$

$$\text{or} \quad = U(t) - \frac{\Delta t}{2}\left(3\left[(\mathrm{U}\cdot\nabla)\mathrm{U}\right](t) - \left[(\mathrm{U}\cdot\nabla)\mathrm{U}\right](t-\Delta t)\right)$$

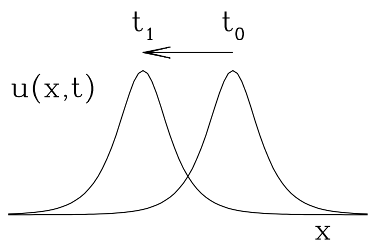
$$(\mathrm{U}(t)\cdot\nabla)\mathrm{U}(t) \sim C\partial_x U(t)$$

Motivates study of methods for solving model of first-order wave equation:

$$\partial_t u = c\partial_x u$$

Analytic solution: traveling wave

$$u(x,t) = u(x+ct,0)$$



General definition of hyperbolic equation in one dimension:

$$\partial_t \mathrm{u} = \partial_x \mathrm{f}(\mathrm{u})$$

First order wave equation has $\mathrm{f}(\mathrm{u}) = c\mathrm{u}$.
Can generalize to $\mathrm{x} \in \mathcal{R}^d$, $\mathrm{u} \in \mathcal{R}^s$, $\mathrm{f}^j : \mathcal{R}^s \to \mathcal{R}^s$

$$\partial_t \mathrm{u}(x_1, \ldots x_d) = \sum_{j=1}^{d} \partial_{x_j} \mathrm{f}^j(\mathrm{u}(x_1, \ldots x_d)) = \nabla \cdot \mathrm{f}(\mathrm{u})$$

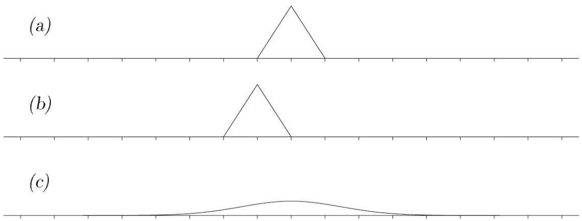
such that partial derivatives $\partial_{x_j} f_s^j$ have certain properties.
Think of $d = 1, 2, 3$ as dimensionality of spatial domain. Think of s as number of variables, e.g. (u, v, w) but also temperature, concentration.

Conservation law for scalar u:

$$\int_\Omega d\Omega \, \partial_t u(x,t) = \int_\Omega d\Omega \, \nabla \cdot \mathrm{f}(u)$$

$$\frac{d}{dt} \int_\Omega d\Omega \, u(x,t) = \int_{\partial\Omega} \mathrm{f}(u) \cdot \mathrm{n}$$

i.e. $\mathrm{f}(u)$ is the flux of u and change in quantity of u in domain is only due to flux in or out through boundary $\partial\Omega$ (no source or sink terms in bulk).



Initial condition

Wave: $\partial_t u = \partial_x u$

Heat: $\partial_t u = \partial_{xx} u$

Hyperbolic PDE of order n is well-posed initial value problem for first $n-1$ derivatives.
For first order

$$\partial_t u = c\partial_x u$$

must specify $u(x, t = 0)$.
For second-order

$$\partial_{tt} u = c^2 \partial_{xx} u$$

must specify initial conditions $u(x, t = 0)$ and $\partial_x u(x, t = 0)$.

Finite difference methods for solving hyperbolic problems

$$\partial_t u = \partial_x u$$

Leapfrog:

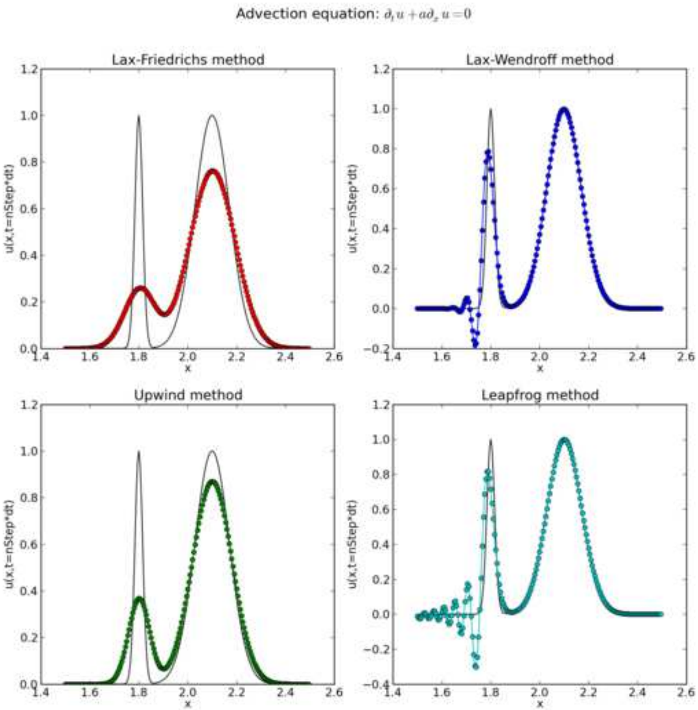
$$u(x, t + \Delta t) = u(x, t - \Delta t) + \frac{\Delta t}{\Delta x}(u(x + \Delta x, t) - u(x - \Delta x, t))$$

Upwind:

$$u(x, t + \Delta t) = u(x, t) + \frac{\Delta t}{\Delta x}(u(x + \Delta x, t) - u(x, t))$$

Lax-Wendroff:

$$u(x, t + \Delta t) = u(x, t) + \frac{\Delta t}{2\Delta x}(u(x + \Delta x, t) - u(x - \Delta x, t)) + \frac{\Delta t^2}{2\Delta x^2}(u(x + \Delta x, t) - 2u(x, t) + u(x - \Delta x, t))$$



Leapfrog suffers from spurious high-wavenumber oscillations.
Lax-Wendroff damps these oscillations.

Accuracy of Leapfrog Method

$$u(x,t+\Delta t) = u(x,t-\Delta t) + \tfrac{\Delta t}{\Delta x}\left(u(x+\Delta x,t) - u(x-\Delta x,t)\right)$$

$$\begin{aligned}
u(x,t-\Delta t) &= \left(u - \Delta t u_t + \tfrac{1}{2}(\Delta t)^2 u_{tt} + \ldots\right)(x,t) \\
\tfrac{\Delta t}{\Delta x} u(x+\Delta x,t) &= \tfrac{\Delta t}{\Delta x}\left(\cancel{u} + \Delta x u_x + \cancel{\tfrac{1}{2}(\Delta x)^2 u_{xx}} + \ldots\right)(x,t) \\
-\tfrac{\Delta t}{\Delta x} u(x-\Delta x,t) &= -\tfrac{\Delta t}{\Delta x}\left(\cancel{u} - \Delta x u_x + \cancel{\tfrac{1}{2}(\Delta x)^2 u_{xx}} + \ldots\right)(x,t)
\end{aligned}$$

$$\textbf{Sum} = \left(u - \Delta t u_t + 2\Delta t u_x + \tfrac{1}{2}(\Delta t)^2 u_{tt} + \ldots\right)(x,t)$$

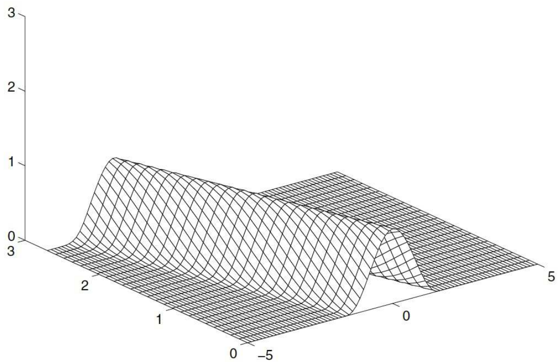
$$u(x,t+\Delta t) = \left(u + \Delta t u_t + \tfrac{1}{2}(\Delta t)^2 u_{tt} + \ldots\right)(x,t)$$

$$\textbf{Error} = 2\Delta t(u_t - u_x) + O(\Delta x,\Delta t)^3 = O(\Delta x,\Delta t)^3$$

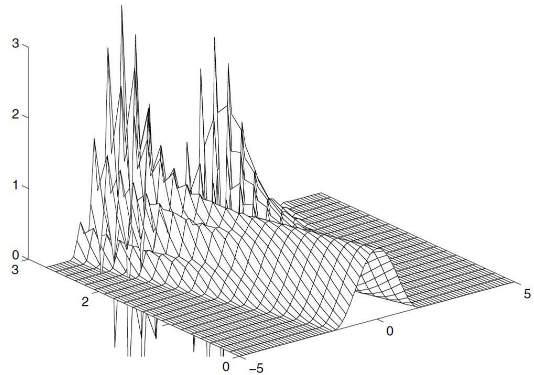
Stability of Leapfrog Method

Leapfrog method used to solve $\partial_t u = \partial_x u$

with initial condition: $u(x, t = 0) = \begin{cases} \cos^2 x & |x| \leq \frac{\pi}{2} \\ 0 & |x| \leq \frac{\pi}{2} \end{cases}$



$$\lambda = \frac{\Delta t}{\Delta x} = 0.9$$



$$\lambda = \frac{\Delta t}{\Delta x} = 1.1$$

Fourier or von Neumann stability analysis: insert $e^{i\xi x}$

$$u(j\Delta x, n\Delta t) = z^n e^{i\xi x_j} \qquad \lambda \equiv \Delta t/\Delta x$$

Leapfrog: $$z^{n+1}e^{i\xi x_j} = z^{n-1}e^{i\xi x_j} + \lambda\left(z^n e^{i\xi x_{j+1}} - z^n e^{i\xi x_{j-1}}\right)$$

$$z = z^{-1} + \lambda\left(e^{i\xi\Delta x} - e^{-i\xi\Delta x}\right)$$

$$z - z^{-1} = 2i\lambda\sin(\xi\Delta x)$$

$$u(x, t+\Delta t) = u(x,t-\Delta t) + \frac{\Delta t}{\Delta x}(u(x+\Delta x, t) - u(x-\Delta x, t))$$

Upwind: $$z^{n+1}e^{i\xi x_j} = z^n e^{i\xi x_j} + \lambda\left(z^n e^{i\xi x_{j+1}} - z^n e^{i\xi x_j}\right)$$

$$z = 1 + \lambda\left(e^{i\xi\Delta x} - 1\right)$$

Lax-Wendroff: $$z^{n+1}e^{i\xi x_j} = z^{n-1}e^{i\xi x_j} + \tfrac{1}{2}z^n\lambda\left(e^{i\xi x_{j+1}} - e^{i\xi x_{j-1}}\right) + \tfrac{1}{2}\lambda^2 z^n\left(e^{i\xi x_{j+1}} - 2e^{i\xi x_j} + e^{i\xi x_{j-1}}\right)$$

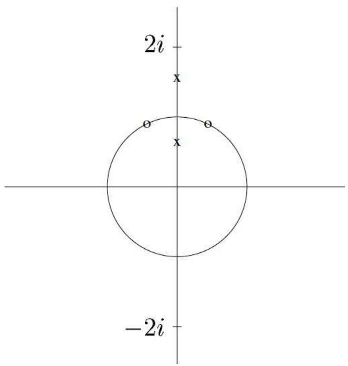
$$z = z^{-1} + \tfrac{1}{2}\lambda\left(e^{i\xi\Delta x} - e^{-i\xi\Delta x}\right) + \tfrac{1}{2}\lambda^2\left(e^{i\xi\Delta x} - 2 + e^{-i\xi\Delta x}\right)$$

$$z = z^{-1} + \lambda i\sin(\xi\Delta x) + \lambda^2(\cosh(\xi\Delta x) - 1)$$

$$z - z^{-1} = 2i\lambda\sin(\xi\Delta x)$$
$$0 = z^2 - 1 - 2i\lambda\sin(\xi\Delta x)$$
$$z = i\lambda\sin(\xi\Delta x) \pm \sqrt{1 - \lambda^2\sin^2(\xi\Delta x)}$$

$$\lambda < 1 \Longrightarrow 1 - \lambda^2\sin^2(\xi\Delta x) > 0 \Longrightarrow a = \sqrt{1 - \lambda^2\sin^2(\xi\Delta x)} \quad \textbf{real}$$
$$|z|^2 = \lambda^2\sin^2(\xi\Delta x) + 1 - \lambda^2\sin^2(\xi\Delta x) = 1$$

$$\lambda > 1 \Longrightarrow 1 - \lambda^2\sin^2(\xi\Delta x) < 0 \Longrightarrow a = \sqrt{1 - \lambda^2\sin^2(\xi\Delta x)} \quad \textbf{imag}$$
$$|z|^2 = (i\lambda\sin(\xi\Delta x) + a)^2 > 1 \textbf{ for values of } \xi\Delta x \textbf{ surrounding } \pi/2$$

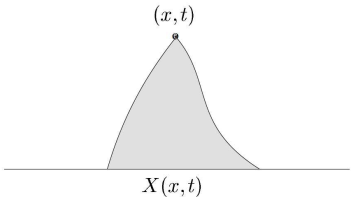


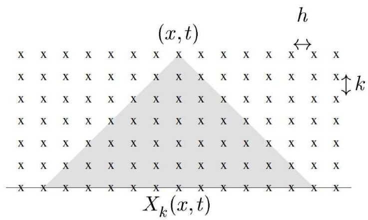
$|z| > 1 \Longrightarrow$ **growth in time**

$\circ \qquad \lambda = 0.9$

$\times \qquad \lambda = 1.1$

Domain of Dependence (Courant, Friedrichs, Lewy = CFL)





continuous system **discretized system**

$X(x,t)$ is the set of all points x where the initial data $u(x,t=0)$ may have some effect on the solution $u(x,t)$.

In parabolic systems such as $u_t = u_{xx}$, information travels infinitely fast. Magnitude of influence of faraway data may decay exponentially with distance but influence will still be present. $X(x,t)$ is the entire real line.

In hyperbolic systems such as $u_t = u_x$ or $u_{tt} = u_{xx}$, perturbations travel at finite speed and so $X(x,t)$ is finite for each x and t. Curves which bound $X(x,t)$ are the characteristic curves.

For convergence of a numerical approximation of a PDE, the continuous domain of dependence must be contained in the limiting numerical domain of dependence as $k = \Delta t \to 0$.