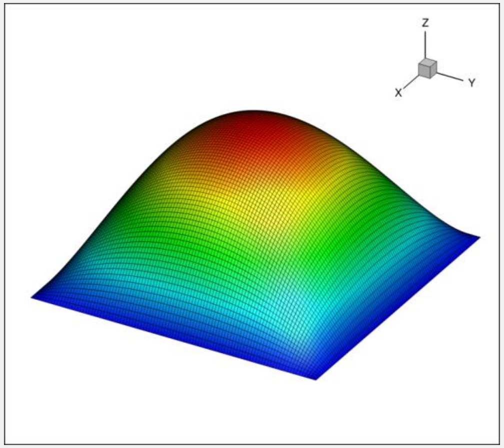
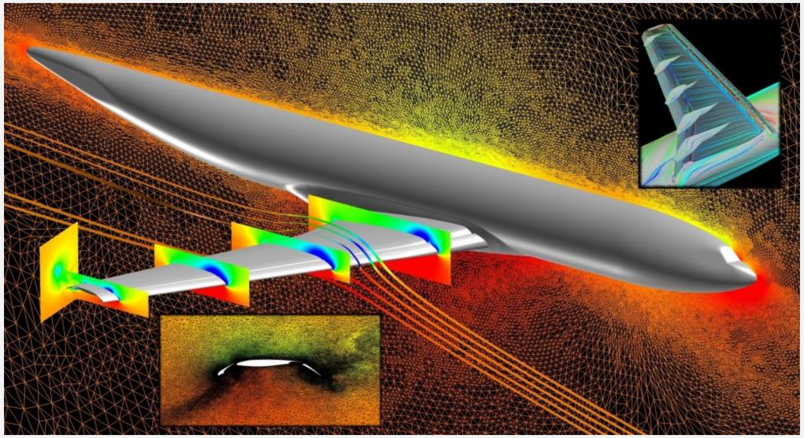
Introduction to PDEs

Greg Byrne

Scholar-In-Resident
Food and Drug Administration





U.S Food and Drug Administration

Basic notions and notations

- A partial differential equation (PDE) is an equation that
  - Has an unknown function depending on at least two variables
  - Contains some partial derivatives of the unknown function

- The following notation will be used in this talk:
- t,x,y,z – are independent variables including time and space coordinates.
- u=u(t,x,…) – are dependent variables (unknown functions) whose partial derivatives are denoted

$$u_t = \frac{\partial u}{\partial t} \qquad u_{tt} = \frac{\partial^2 u}{\partial t^2} \qquad u_{xy} = \frac{\partial^2 u}{\partial x \partial y}$$

Methods and techniques for solving PDEs

- Separation of variables
- Integral transforms
- Change of coordinates
- Transformation of the dependent variable
- Numerical Methods. A PDE is changed into a system of difference equations that can be solved by iterative techniques (Finite Difference Methods). Solutions can also be approximated using polynomial functions (e.g. Finite Element Methods).
- Perturbation Methods
- Impulse-response technique
- Integral equations
- Variational methods
- Eigenfunction expansion

Well-posed and ill-posed problems

- An initial-boundary-value problem is well-posed if:
  - a solution exists
  - the solution is unique
  - the solution depends continuously on the data (boundary and/or initial conditions). *Small changes in the data should cause only small changes in the solution.*
- Problems which do not fulfill these criteria are ill-posed.
- Well posed problems have a good chance of being solved numerically by a stable algorithm.
  - Unavoidable small errors in initial and boundary data produce only slight errors in the computed solution leading to useful results.

Basic Classification of PDEs

- Order of the PDE. The order of a PDE is the order of the highest partial derivative in the equation.
  - First order: $u_t = u_x$
  - Second order: $u_t = u_{xx}$ $\quad u_{xy} = 0$
  - Third order: $u_t + uu_{xxx} = sin(x)$
  - Fourth order: $u_{tt} = u_{xxxx}$

Basic Classification of PDEs

- Order of the PDE. The order of a PDE is the order of the highest partial derivative in the equation.
- Number of variables. PDEs may be classified by the number of their independent variables, that is, the number of variables the unknown function depends on.
  - PDE in two variables: $u_t = u_{xx} \qquad (u = u(t,x))$
  - PDE in three variables: $u_t = u_{rr} + \frac{1}{r}u_r + \frac{1}{r^2}u_{\theta\theta} \quad (u = u(t,r,\theta))$
  - PDE in four variables: $u_t = u_{xx} + u_{yy} + u_{zz} \qquad (u = u(t,x,y,z))$

Basic Classification of PDEs

- Order of the PDE. The order of a PDE is the order of the highest partial derivative in the equation.
- Number of variables. PDEs may be classified by the number of their independent variables, that is, the number of variables the unknown function depends on.
- Linearity. A PDE is linear if the dependent variable and all its derivatives appear in a linear fashion.
  - linear: $u_{tt} + exp(-t)u_{xx} = sin(t)$ $\quad xu_{xx} + yu_{yy} = 0$
  - nonlinear: $uu_{xx} + u_t = 0$ $\quad u_x + u_y + u^2 = 0$

Basic Classification of PDEs

- **Order of the PDE.** The order of a PDE is the order of the highest partial derivative in the equation.
- **Number of variables.** PDEs may be classified by the number of their independent variables, that is, the number of variables the unknown function depends on.
- **Linearity.** A PDE is linear if the dependent variable and all its derivatives appear in a linear fashion.
- **Coefficients.** PDEs can contain constant or variable coefficients (i.e. if at least one of the coefficients is a function of some independent variable)
  - constant coefficients: $u_{tt} + 5u_{xx} - 3u_{xy} = 0$
  - variable coefficients: $u_t + exp(-t)u_{xx} = 0$

Basic Classification of PDEs

- Order of the PDE. The order of a PDE is the order of the highest partial derivative in the equation.
- Number of variables. PDEs may be classified by the number of their independent variables, that is, the number of variables the unknown function depends on.
- Linearity. A PDE is linear if the dependent variable and all its derivatives appear in a linear fashion.
- Coefficients. PDEs can contain constant or variable coefficients (i.e. if at least one of the coefficients is a function of some independent variable)
- Homogeneity. A PDE is homogenous if the free term (right-hand side) is zero.

(homogenous) $u_{tt} - u_{xx} = 0$

(non-homogenous) $u_{tt} - u_{xx} = x^2 sin(t)$

Types of second-order linear PDEs

- A second-order linear PDE in two variables can be written in the following general form:

$$Au_{xx} + Bu_{xy} + Cu_{yy} + Du_x + Eu_y + Fu = G$$

where A,B,C,D,E,F are coefficients and G is a non-homogenous term.

- These PDEs are classified into three types whose mathematical solutions are quite different:
  - Hyperbolic: $B^2 - 4AC > 0 \; (e.g. \; u_{tt} - u_{xx} = 0)$
  - Parabolic: $B^2 - 4AC = 0 \; (e.g. \; u_t - u_{xx} = 0)$
  - Elliptic: $B^2 - 4AC < 0 \; (e.g. \; u_{xx} - u_{yy} = 0)$

Types of second-order linear PDEs

- A second-order linear PDE in two variables can be written in the following general form:

$$Au_{xx} + Bu_{xy} + Cu_{yy} + Du_x + Eu_y + Fu = G$$

where A,B,C,D,E,F are coefficients and G is a non-homogenous term.

- These PDEs are classified into three types whose mathematical solutions are quite different.
- The three major classifications of linear PDEs describe physical problems into three basic types:
  - Vibrating systems and wave propagation (hyperbolic)
  - Heat flow and diffusion processes (parabolic)
  - Stead-state phenomena (elliptic)

Some Basic PDEs

- The heat equation (parabolic)

$$u_t = c u_{xx}$$

- The wave equation (hyperbolic)

$$u_{tt} = c^2 u_{xx}$$

- The Poisson equation (elliptic)

$$\nabla^2 u = f(x, y) \qquad \nabla^2 = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}$$

Solving PDEs: Recipe List

- A PDE
- Solution domain (regular, irregular)
- Boundary conditions
  - **Dirichlet B.C.** Specify u(x,y,...) on boundaries
  - **Neumann B.C.** Specify normal gradient of u(x,y,...) on boundaries.
- Initial values
- Stable and convergent numerical algorithm

Numerical Derivatives: Finite Differences

- Introduce finite difference

Forward difference

$$f'(a) \approx \frac{f(a+h)-f(a)}{h} + O(h)$$

Backward difference

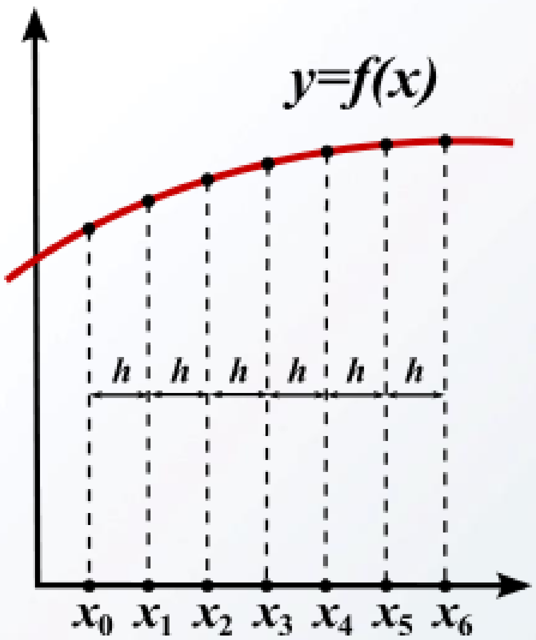
$$f'(a) \approx \frac{f(a)-f(a-h)}{h} + O(h)$$

Central difference

$$f'(a) \approx \frac{f(a+h)-f(a-h)}{2h} + O(h^2)$$

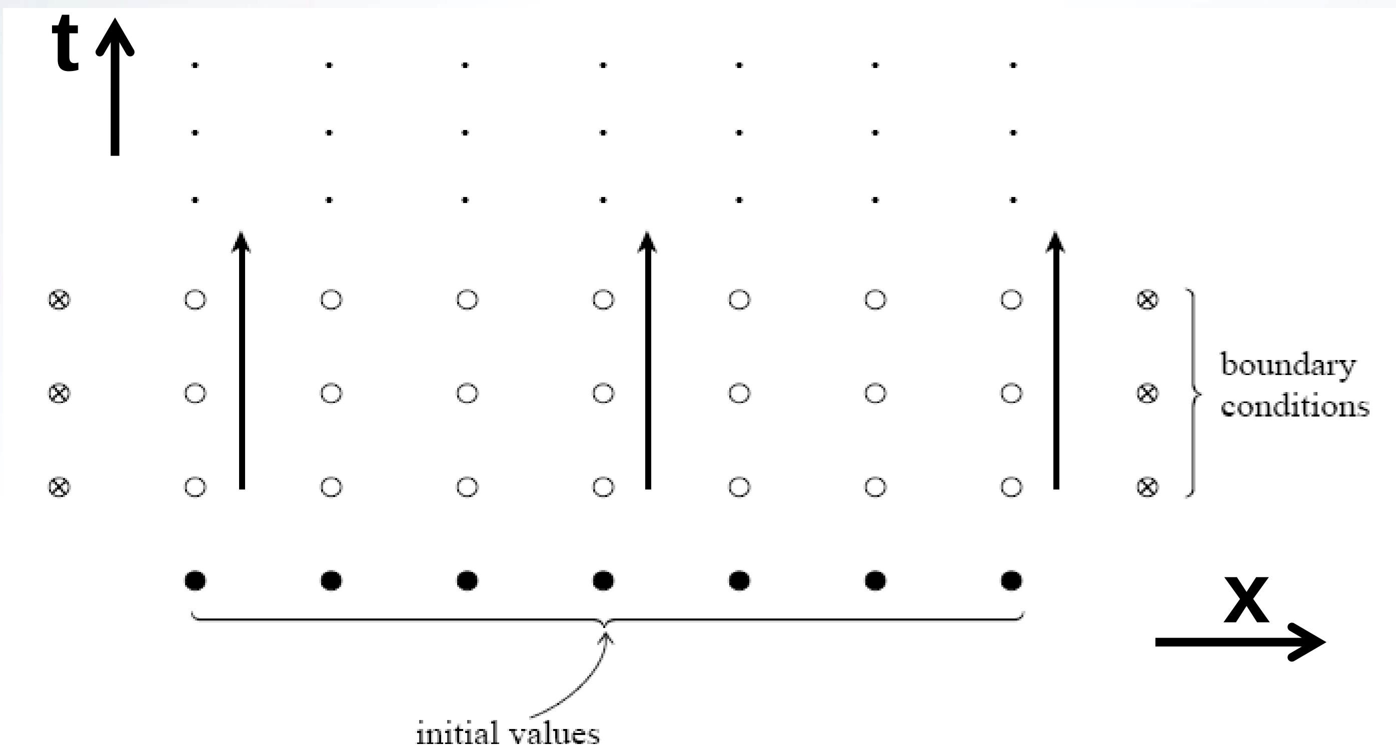
Central difference (2nd order)

$$f''(a) \approx \frac{f(a+h)-2f(a)+f(a-h)}{h^2} + O(h^2)$$



Heat Equation Recipe

- The PDE: $u_t = cu_{xx}$
- The domain: 1-d rod
- Boundary conditions (dirichlet) and initial values.
- Numerical algorithm (tbd)



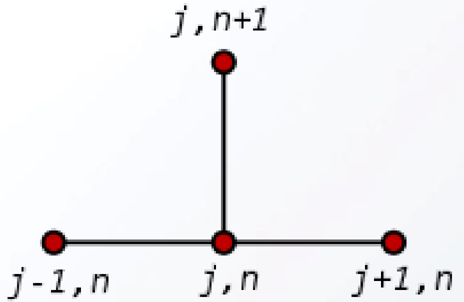
Heat Equation: Explicit Method

- Use forward difference for time derivative and second-order central difference for space derivative:

$$u_t = c u_{xx}$$

where c=1

$$\frac{u_j^{n+1} - u_j^n}{k} = \frac{u_{j+1}^n - 2u_j^n + u_{j-1}^n}{h^2}.$$



- Iterate equation: $u_j^{n+1} = (1-2r)u_j^n + r u_{j-1}^n + r u_{j+1}^n$

  where $r = k/h^2.$

  – Knowing values at time step n allows you to compute values at time step n+1

- This method is numerically stable and convergent when $r \le 1/2$ with numerical errors proportional to

$$\Delta u = O(k) + O(h^2)$$

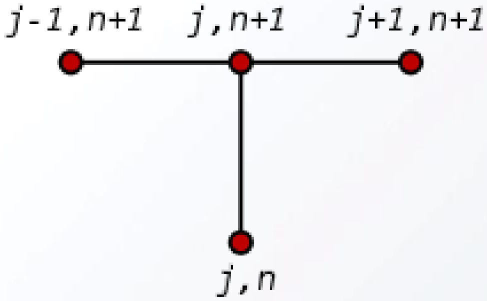
Heat Equation: Implicit Method

- Use backward difference at time n+1 and a second-order central difference for space derivative:

$$u_t = c u_{xx}$$

where c=1

$$\frac{u_j^{n+1} - u_j^n}{k} = \frac{u_{j+1}^{n+1} - 2u_j^{n+1} + u_{j-1}^{n+1}}{h^2}.$$



- Solve linear eqs: $(1+2r)u_j^{n+1} - r u_{j-1}^{n+1} - r u_{j+1}^{n+1} = u_j^n$

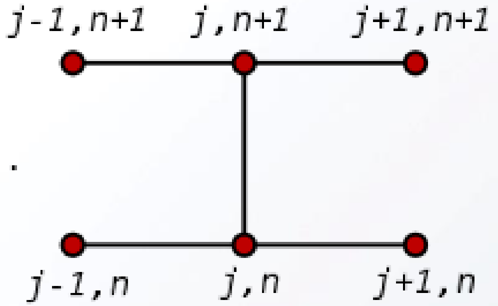
  where $r = k/h^2.$
  - Solving linear system of equations at each step is more numerically intensive than explicit methods.
- This method is always numerically stable and convergent with numerical errors proportional to

$$\Delta u = O(k) + O(h^2)$$

Heat Equation: Crank-Nicolson Method

- Use central difference at time n+1 and a second-order central difference for space derivative:

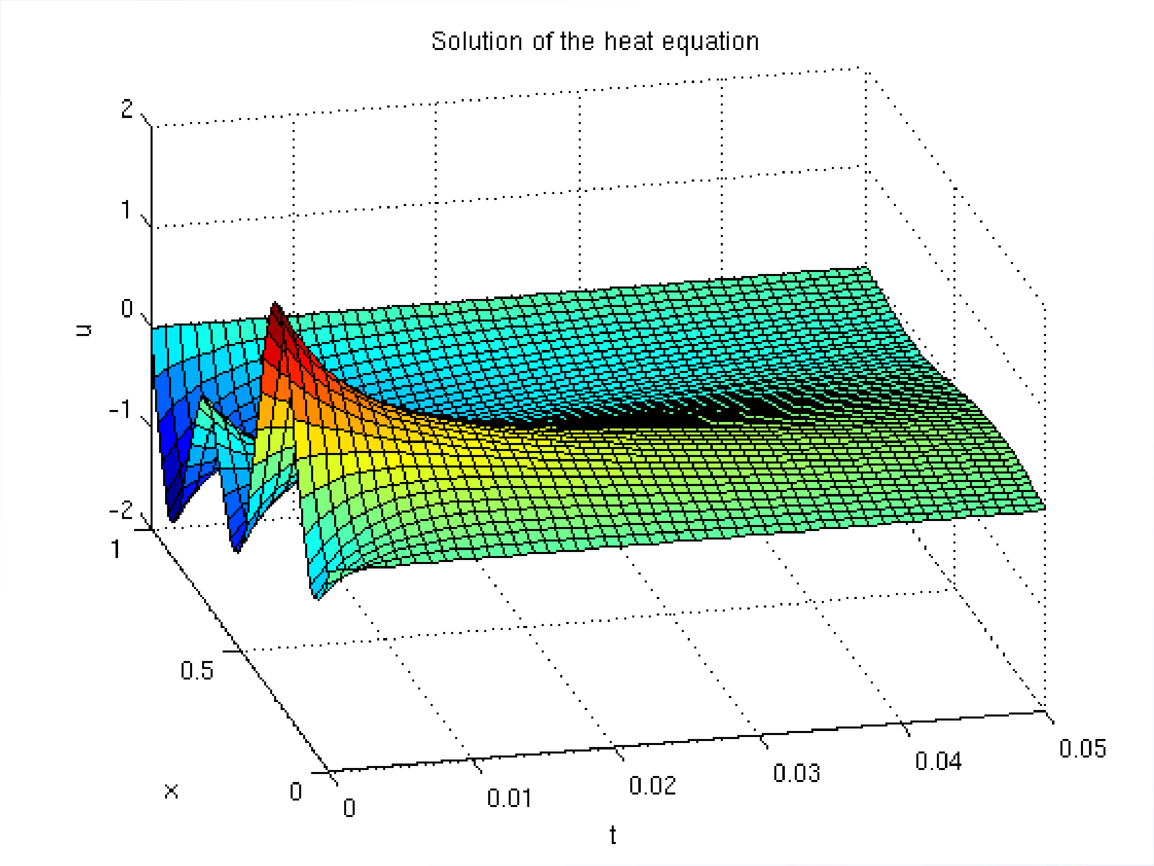
$$\frac{u_j^{n+1} - u_j^n}{k} = \frac{1}{2}\left(\frac{u_{j+1}^{n+1} - 2u_j^{n+1} + u_{j-1}^{n+1}}{h^2} + \frac{u_{j+1}^n - 2u_j^n + u_{j-1}^n}{h^2}\right).$$



- Solve linear eqs: $(2+2r)u_j^{n+1} - ru_{j-1}^{n+1} - ru_{j+1}^{n+1} = (2-2r)u_j^n + ru_{j-1}^n + ru_{j+1}^n$
  where $r = k/h^2$.
  - Solving linear system of equations at each step is more numerically intensive than explicit methods but more accurate.
- This method is <u>always</u> numerically stable and convergent with numerical errors proportional to

$$\Delta u = O(k^2) + O(h^2).$$

Heat Equation: Example Solution



Numerical Derivatives: Spectral Methods

In space

$$\hat{u}(x) = \sum_{k=-\infty}^{\infty} u_k e^{ikx}$$

$$\frac{\partial^n u}{\partial x^n} = \sum_{k=-\infty}^{\infty} (ik)^n \hat{u}(x)$$

In time

$$\hat{u}(x,t) = \sum_{k=-\infty}^{\infty} u_k(t) e^{ikx}$$

$$\frac{\partial^n u}{\partial t^n} = \sum_{k=-\infty}^{\infty} \frac{d^n u_k}{dt^n} e^{ikx}$$

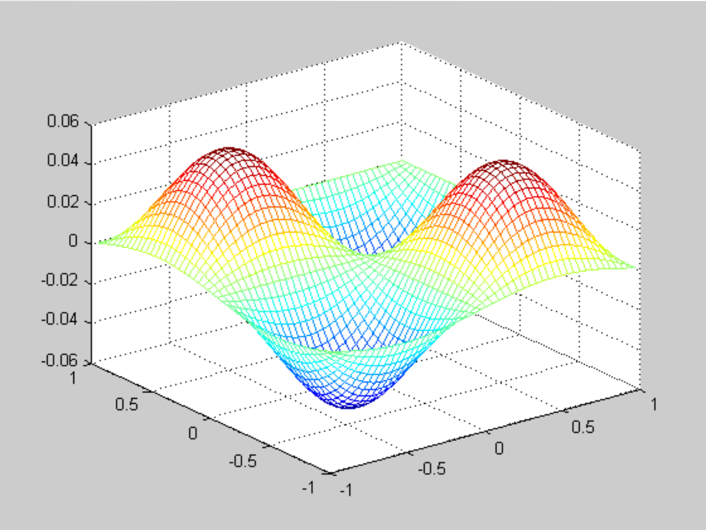
n	(ik)^n	Feature
1	ik	Propagation (no dispersion)
2	- k^2	Decay
3	- ik^3	Propagation (dispersion)
4	k^4	Growth

Poisson: Spectral Method

- Apply the fourier transform to both sides of the equation

$$\nabla^2 u = f(x,y) \quad -(k_x^2 + k_y^2)\hat{u}(x,y) = \hat{f}(x,y)$$

- Take the inverse Fourier transform of $\hat{u}(x,y) = \dfrac{\hat{f}(x,y)}{-(k_x^2 + k_y^2)}$



Sample solution for:

$$f(x,y) = sin(\pi x) + sin(\pi y)$$

Reaction Diffusion Equations: Laplacian

$$\partial_t \mathbf{w} = F(\mathbf{w}) = D\nabla^2 \mathbf{w} + g(\mathbf{w})$$

- Finite Difference 5-point
  - fast

$$\nabla^2 V_{i,j} \approx \frac{1}{dx^2}\begin{pmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{pmatrix}$$

- Finite Difference 9-point
  - preserves symmetry

$$\nabla^2 V_{i,j} \approx \frac{1}{dx^2}\begin{pmatrix} 1/6 & 2/3 & 1/6 \\ 2/3 & -10/3 & 2/3 \\ 1/6 & 2/3 & 1/6 \end{pmatrix}$$

- Spectral

$$\nabla^2 V = FFT^{-1}(-(k_x^2 + k_y^2)\hat{V})$$

Reaction Diffusion Equations: Time Integration

$$\partial_t \mathbf{w} = F(\mathbf{w}) = D\nabla^2\mathbf{w} + g(\mathbf{w})$$

- Lots of time integrators available to solve the PDE
- One family of integrators involve Runge-Kutta (1,2,4) methods
  - For example the Euler (RK1):

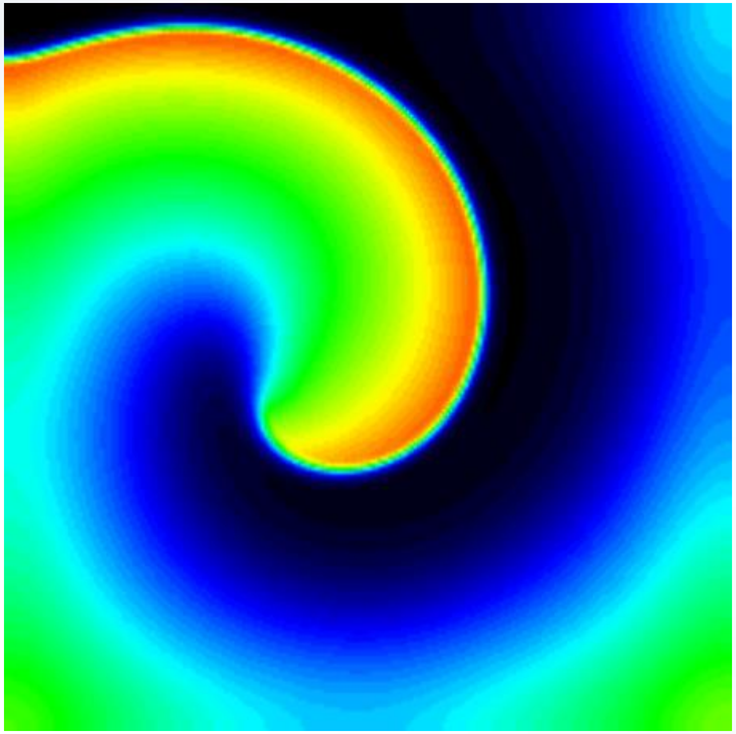
$$\mathbf{w}_{t+1} = \mathbf{w}_t + \Delta t F(\mathbf{w})$$

  - RK 2,4 are multi-step and therefore more computationally expensive but more accurate.

Reaction Diffusion Equation: Putting it all together

- The PDE (choose your cardiac cell model)
- Solution domain: regular, irregular (phase field)
- Boundary conditions are Neumann: no flux or periodic
- Initial values
- Stable and convergent numerical algorithm. Your pick!

Questions?



EXTRA: Crank-Nicolson

- Consider the PDE: $$\frac{\partial u}{\partial t} = F\left(u,\, x,\, t,\, \frac{\partial u}{\partial x},\, \frac{\partial^2 u}{\partial x^2}\right)$$

- The CN method is a combination of the forward Euler method at time step n and backward Euler method at step n+1:

$$\frac{u_i^{n+1} - u_i^n}{\Delta t} = F_i^n\left(u,\, x,\, t,\, \frac{\partial u}{\partial x},\, \frac{\partial^2 u}{\partial x^2}\right) \qquad \text{(forward Euler)}$$

$$\frac{u_i^{n+1} - u_i^n}{\Delta t} = F_i^{n+1}\left(u,\, x,\, t,\, \frac{\partial u}{\partial x},\, \frac{\partial^2 u}{\partial x^2}\right) \qquad \text{(backward Euler)}$$

$$\frac{u_i^{n+1} - u_i^n}{\Delta t} = \frac{1}{2}\left[F_i^{n+1}\left(u,\, x,\, t,\, \frac{\partial u}{\partial x},\, \frac{\partial^2 u}{\partial x^2}\right) + F_i^n\left(u,\, x,\, t,\, \frac{\partial u}{\partial x},\, \frac{\partial^2 u}{\partial x^2}\right)\right] \qquad \text{(Crank–Nicolson).}$$