ODE and PDE Stability Analysis

COS 323

Last Time

- Finite difference approximations
- Review of finite differences for ODE BVPs
- PDEs
- Phase diagrams
- Chaos

Today

- Stability of ODEs
- Stability of PDEs
- Review of methods for solving large, sparse systems
- Multi-grid methods

Reminders

- Homework 4 due next Tuesday
- Homework 5, final project proposal due Friday December 17
- Final project: groups of 3-4 people

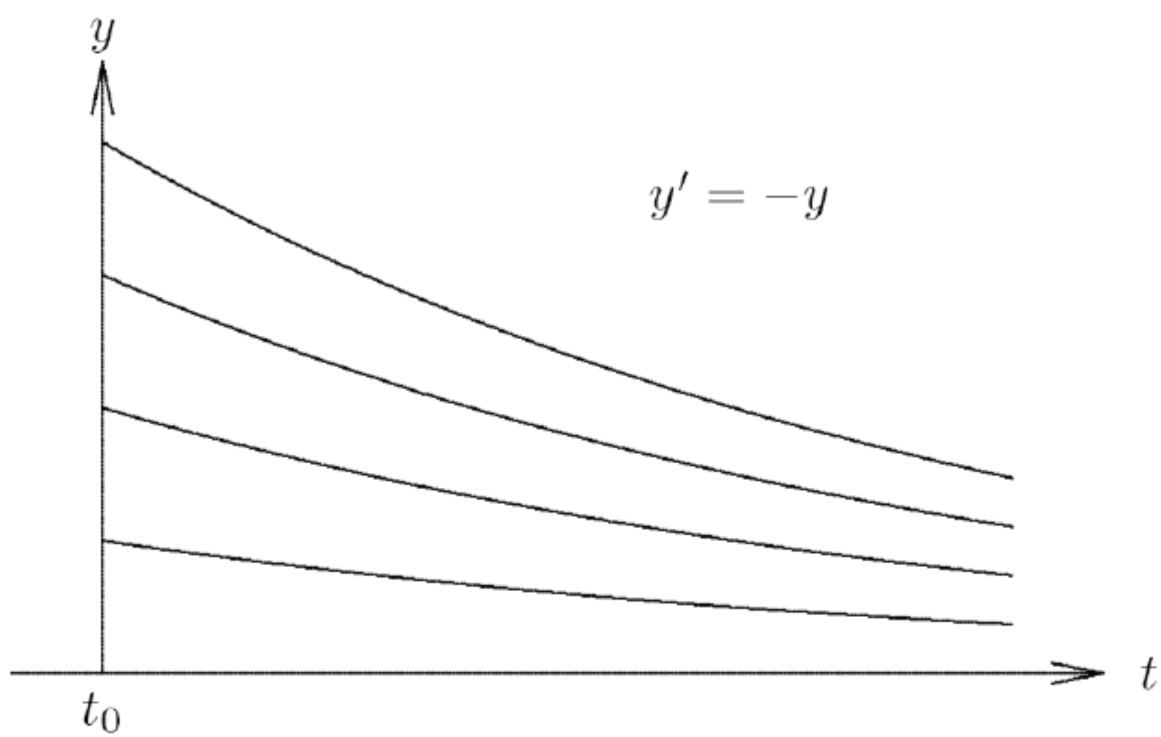
Stability of ODE

A solution of the ODE $y' = f(t, y)$ is stable if for every $\varepsilon > 0$ there is a $\delta > 0$ st if $\hat{y}(t)$ satisfies the ODE and $\left\|\hat{y}(t_0) - y(t_0)\right\| \leq \delta$ then $\left\|\hat{y}(t) - y(t)\right\| \leq \varepsilon$ for all $t \geq t_0$

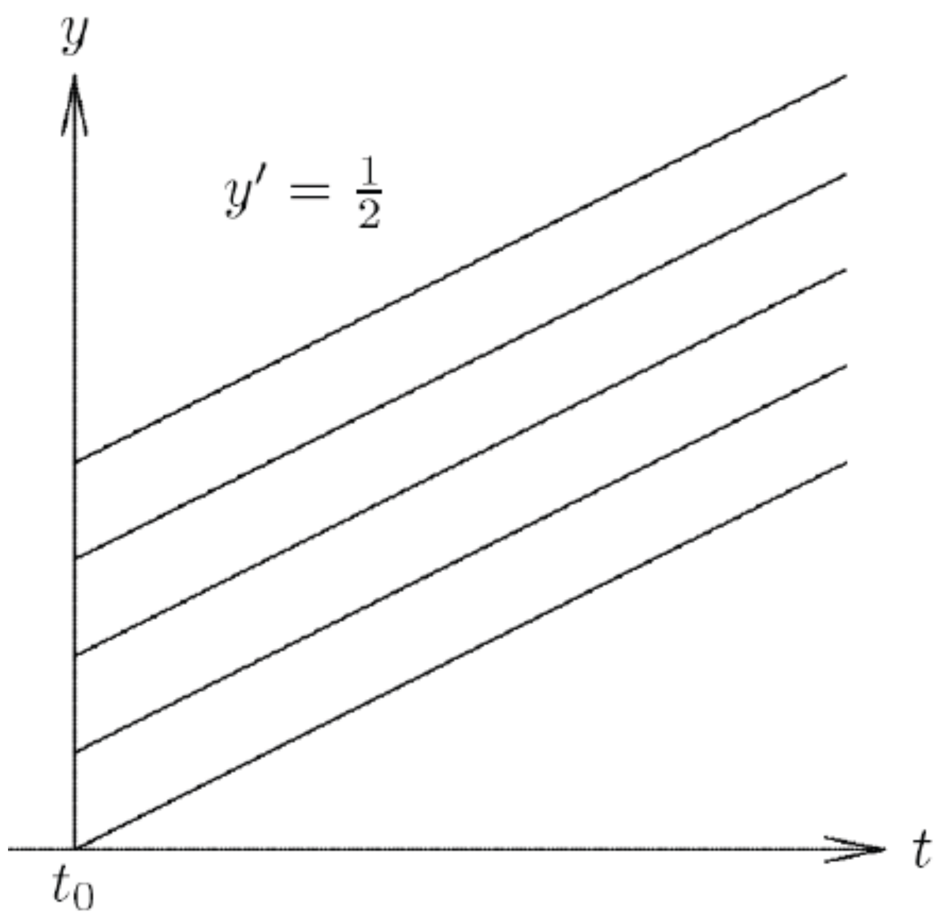
- i.e., rules out exponential divergence if initial value is perturbed

- **asymptotically stable solution:**

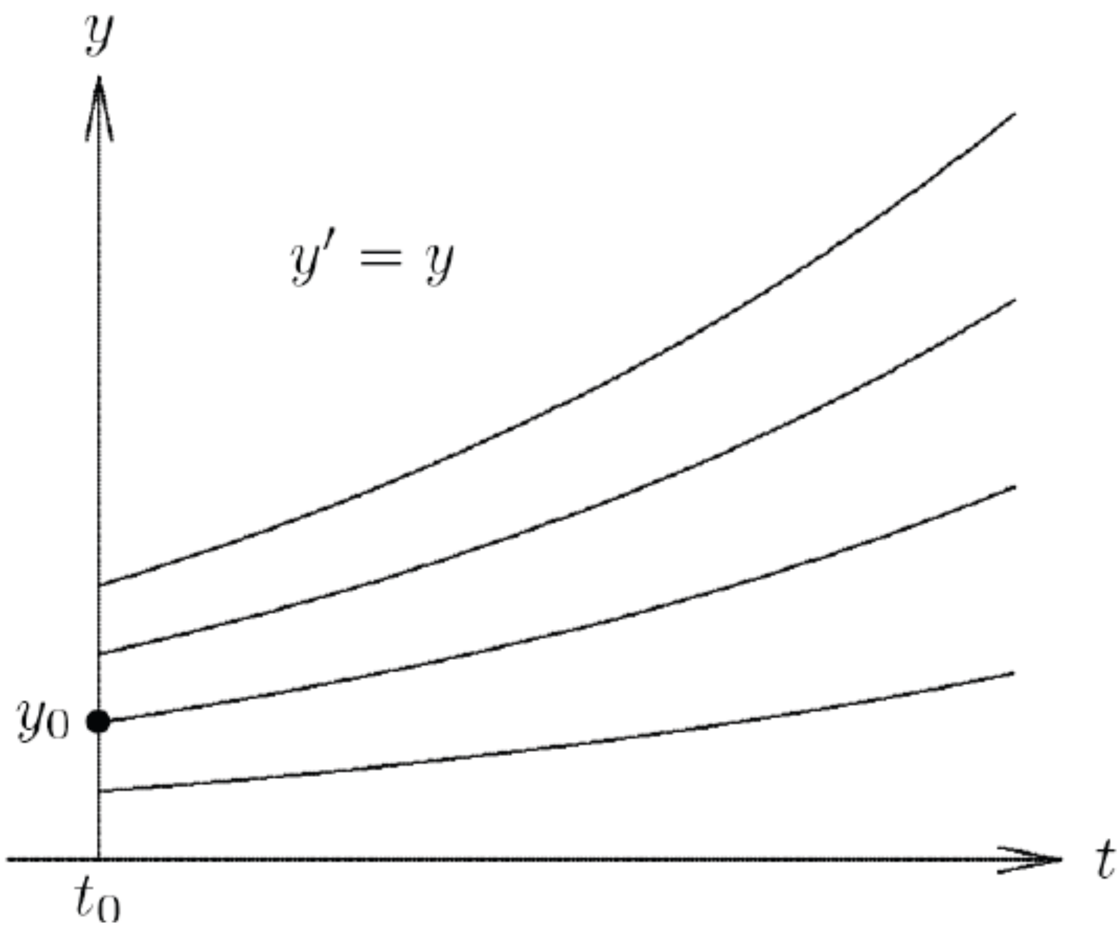
$$\left\|\hat{y}(t) - y(t)\right\| \rightarrow 0 \text{ as } t \rightarrow 0$$



- stable but not asymptotically so:



- unstable:



Determining stability

- General case: $\mathbf{y}' = \mathbf{f}(t, \mathbf{y})$
- Simpler: linear, homogeneous system:

  $$\mathbf{y}' = \mathbf{A}\mathbf{y}$$

- Even simpler: $y' = \lambda y$

$y' = \lambda y$

- Solution: $y(t) = y_0 e^{\lambda t}$
- If $\lambda > 0$: exponential divergence : every solution is unstable
- If $\lambda < 0$: every solution is asymptotically stable
- If λ complex:
  - $e^{\lambda t} = e^{at} (\cos(bt) + i \sin(bt))$
  - $\mathrm{Re}(\lambda)$ is a. This is oscillating component multiplied by a real amplification factor.
  - $\mathrm{Re}(\lambda) > 0$: All unstable; $\mathrm{Re}(\lambda) < 0$: All stable.

Stability: Linear system

- **y' = Ay**
- if **A** is diagonalizable → eigenvectors are linearly independent

$$y_0 = \sum_{i=1}^{n} \alpha_i \mathbf{u}_i \text{ where } \mathbf{u}_i \text{ are eigenvectors of A}$$

$$y(t) = \sum_{i=1}^{n} \alpha_i \mathbf{u}_i e^{\lambda_i t} \text{ is a solution satisfying initial condition}$$

- Component by component: if $\mathrm{Re}(\lambda_i) > 0$ then growing, $\mathrm{Re}(\lambda_i) < 0$ decaying; $\mathrm{Re}(\lambda_i) = 0$ oscillating
- Non-diagonalizable: requires all $\mathrm{Re}(\lambda_i) <= 0$, and $\mathrm{Re}(\lambda_i) < 0$ for any non-simple eigenvalue

Stability with Variable Coefficients

- $y'(t) = A(t)\, y(t)$
- Signs of eigenvalues may change with t, so eigenvalue analysis hard

Stability, in General

- y' = f(t, y)
- Can linearize ODE using truncated Taylor Series:

  $z' = J_f(t, y(t))z$

  where J_f is Jacobian of f with respect to y

  i.e., $\left\{J_f(t,y)\right\}_{ij} = \dfrac{\partial f_i(t,y)}{\partial y_j}$
- If autonomous, then eigenvalue analysis yields same results as for linear ODE; otherwise, difficult to reason about eigenvalues
- NOTE: J_f evaluated at certain value of y_0 (i.e., for a particular solution): so changing y_0 may change stability properties

Summary so far

- A solution to an ODE may be stable or unstable, regardless of method used to solve it
- May be difficult to analyze for non-linear, non-homogenous ODEs
- $y' = \lambda y$ is a good proxy for understanding stability of more complex systems, where λ functions like the eigenvalues of J_f

Stability of ODE vs Stability of Method

- Stability of **ODE solution**: Perturbations of solution do not diverge away over time
- Stability of a **method**:
  - Stable if small perturbations do not cause the solution to diverge from each other without bound
  - Equivalently: Requires that solution at any fixed time t remain bounded as $h \to 0$ (i.e., # steps to get to t grows)
- How does stability of method interact with stability of underlying ODE?
  - ODE may prevent convergence (e.g., $\lambda > 0$)
  - Method may be unstable even when ODE is stable
  - ODE can determine step size h allowed for stability, for a given method

Stability of Euler's Method

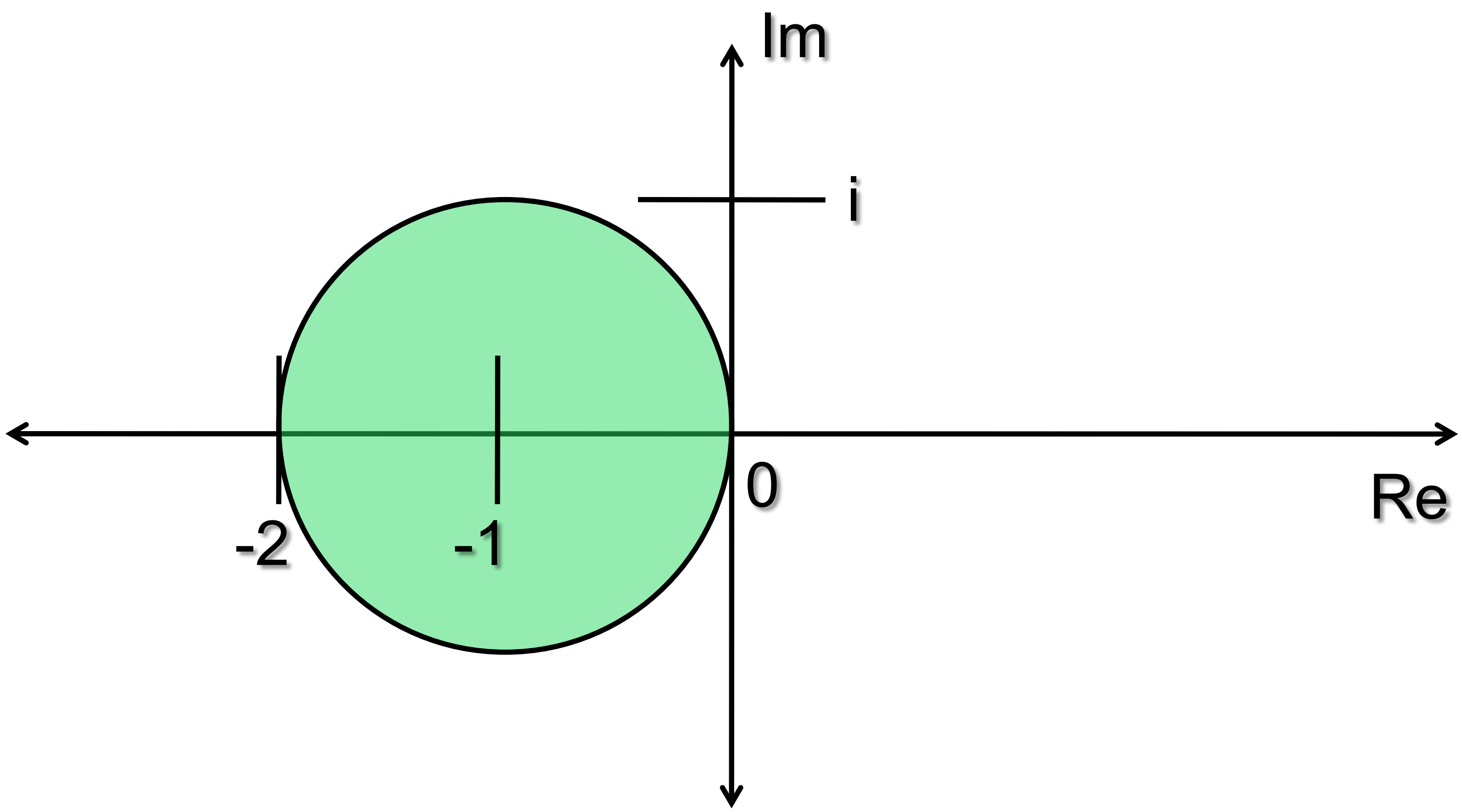
- $y' = \lambda y$: Solution is $y(t) = y_0 e^{\lambda t}$
- Euler's method: $y_{k+1} = y_k + h\lambda y_k$
- $y_{k+1} = (1 + h\lambda)y_k$
- Significance?

  $y_k = (1 + h\lambda)^k y_0$

- $(1 + h\lambda)$ is growth factor
- If $|1 + h\lambda| <= 1$: Euler's is stable
- If $|1 + h\lambda| > 1$: Euler's is unstable

Stability region for Euler's method, y' = λy

- *h*λ must be in circle of radius 1 centered at -1:



i.e., For λ < 0, stable only if h <= -2/λ; **can be unstable even when ODE stable**

Stability for Euler's method, general case

$$e_{k+1} = (\mathbf{I} + h_k \bar{J}_f) e_k + l_{k+1}$$

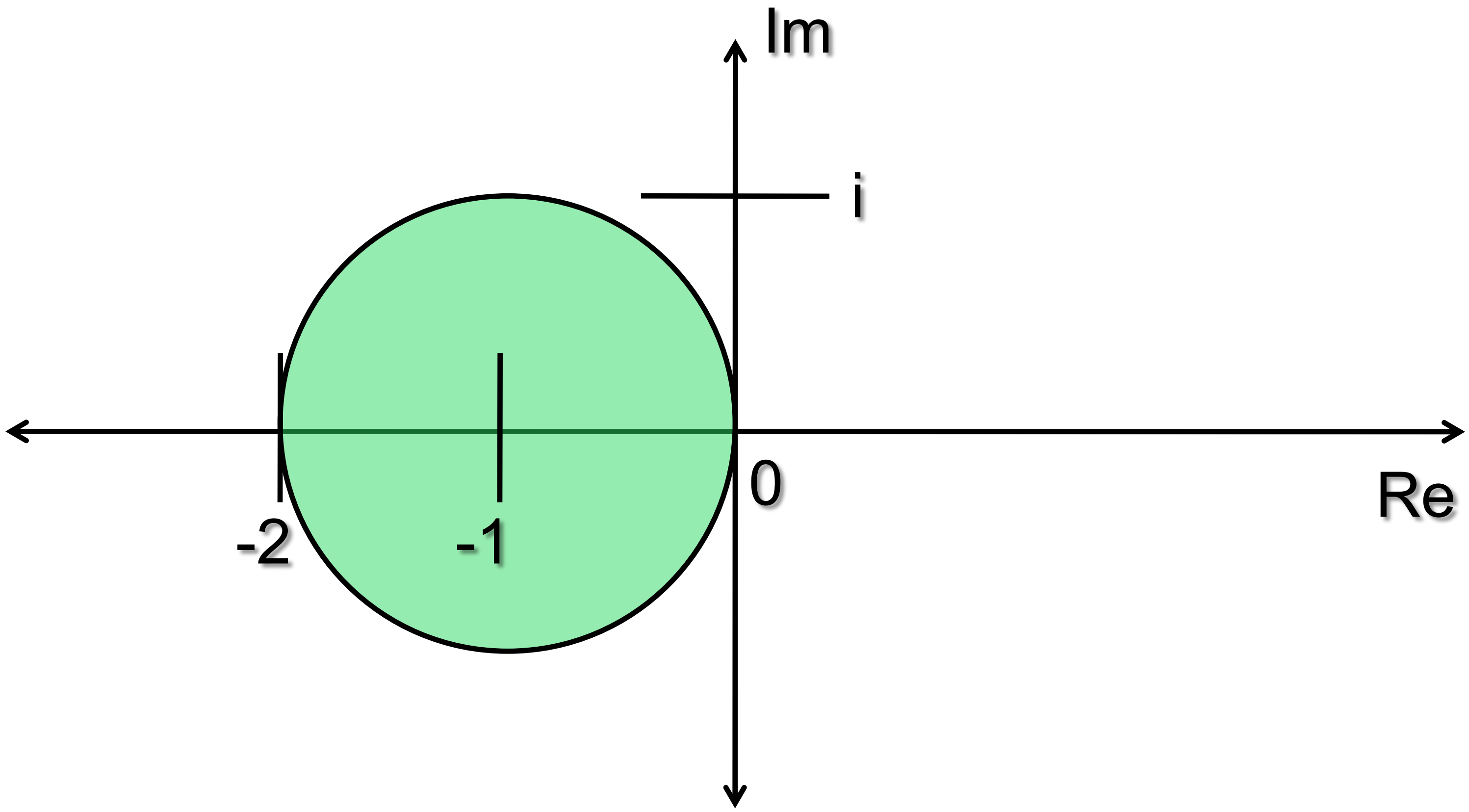
$$\text{where } \bar{J}_f = \int_0^1 J_f(t_k, \alpha y_k + (1-\alpha) y(t_k)\, d\alpha$$

- Growth factor: $\mathbf{I} + h_k \bar{J}_f$
  - Compare to |1 + hλ|
- Stable if spectral radius $\rho(\mathbf{I} + h_k \bar{J}_f) \leq 1$
  - Satisfied if all eigenvalues of $h_k \bar{J}_f$

    lie inside the circle

Stability region for Euler's method, y' = f(t, y)

- Eigenvalues of $h_k \bar{J}_f$ inside



Discussion: Euler's Method

- Stability depends on h, J_f
- Haven't mentioned accuracy at all
- Accuracy is O(h)
  - Can always decrease h without penalty if λ real

Backward Euler

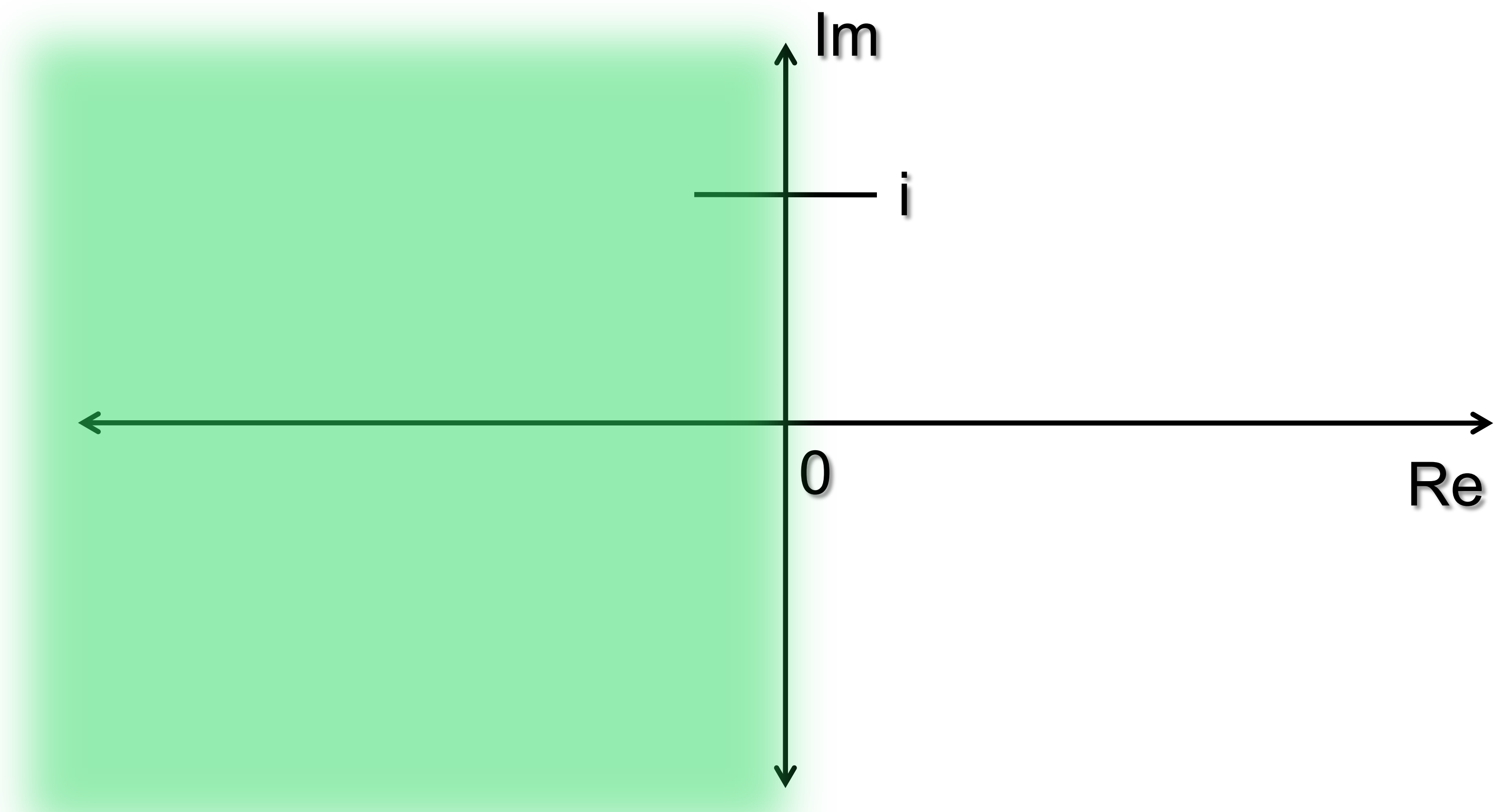
- $y' = \lambda y$
- $y_{k+1} = y_k + h\lambda y_{k+1}$
- $(1-h\lambda)y_{k+1} = y_k$

$$y_k = \left(\frac{1}{1-h\lambda}\right)^k y_0$$

so stability requires $\left|\frac{1}{1-h\lambda}\right| \leq 1$

Stability Region for Backward Euler, $y' = \lambda y$

- Region of stability: $h\lambda$ in left half of complex plane:



i.e., any $h > 0$ when $\text{Re}(\lambda) < 0$

Stability for Backward Euler, general case

- Amplification factor is $(\mathbf{I} - h\mathbf{J}_f)^{-1}$
- Spectral radius < 1 if eigenvalues of $h\mathbf{J}_f$ *outside* circle of radius 1 centered at one
- i.e., **if** solution is stable, then Backward Euler is stable for any positive step size: **unconditionally stable**
- Step size choice can manage efficiency vs accuracy without concern for stability
  - Accuracy is still O(h)

Stability for Trapezoid Method

$$y_{k+1} = y_k + h(\lambda y_k + \lambda y_{k+1})/2$$

$$y_k = \left(\frac{1 + h\lambda/2}{1 - h\lambda/2}\right)^k y_0$$

$$\text{so stable if } \left|\frac{1 + h\lambda/2}{1 - h\lambda/2}\right| \leq 1$$

$$(\text{holds for any h} > 0 \text{ when } \mathrm{Re}(\lambda) < 0)$$

- i.e., **unconditionally stable**
- In general: Amplification factor =

  $$(\mathbf{I} + \tfrac{1}{2}h\mathbf{J}_f)(\mathbf{I} - \tfrac{1}{2}h\mathbf{J}_f)^{-1}$$

  spectral radius $<$ 1 if eigenvalues of $h\mathbf{J}_f$ lie in left half of plane

Implicit methods

- Generally larger stability regions than explicit methods
- Not always unconditionally stable
  - i.e., step size does matter sometimes

Stiffness and Stability

- for $y' = \lambda y$:
- stiff over interval b – a if

  $(b - a)\,\mathrm{Re}(\lambda) \ll -1$

  i.e., λ may be negative but large in magnitude (a stable ODE)

  Euler's method stability requires $|1 + h\lambda| < 1$

  therefore requires VERY small h

  Backward Euler fine: any step size still OK (see graph)

Conditioning of Boundary Value Problems

- Method does not travel "forward" (or "backward") in time from an initial condition
- No notion of asymptotically stable or unstable
- Instead, concern for interplay between solution modes and boundary conditions
  - growth forward in time is limited by boundary condition at b
  - decay forward in time is limited by boundary condition at a
- See "Boundary Value Problems and Dichotomic Stability," England & Mattheij, 1988

PDEs

Finite Difference Methods: Example

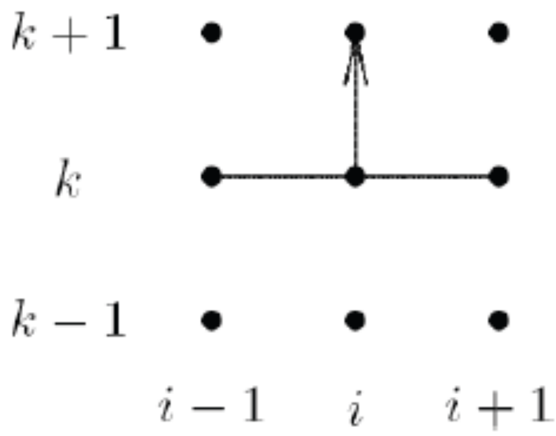
- Consider heat equation

$$u_t = c\, u_{xx}, \qquad 0 \le x \le 1, \qquad t \ge 0$$

with initial and boundary conditions

$$u(0, x) = f(x), \qquad u(t, 0) = \alpha, \qquad u(t, 1) = \beta$$

$$\frac{u_i^{k+1} - u_i^k}{\Delta t} = c\, \frac{u_{i+1}^k - 2u_i^k + u_{i-1}^k}{(\Delta x)^2}$$



Example, Continued

- Finite difference method yields recurrence relation:

$$u_i^{k+1} = u_i^k + c\,\frac{\Delta t}{(\Delta x)^2}\left(u_{i+1}^k - 2u_i^k + u_{i-1}^k\right), \quad i = 1, \ldots, n$$

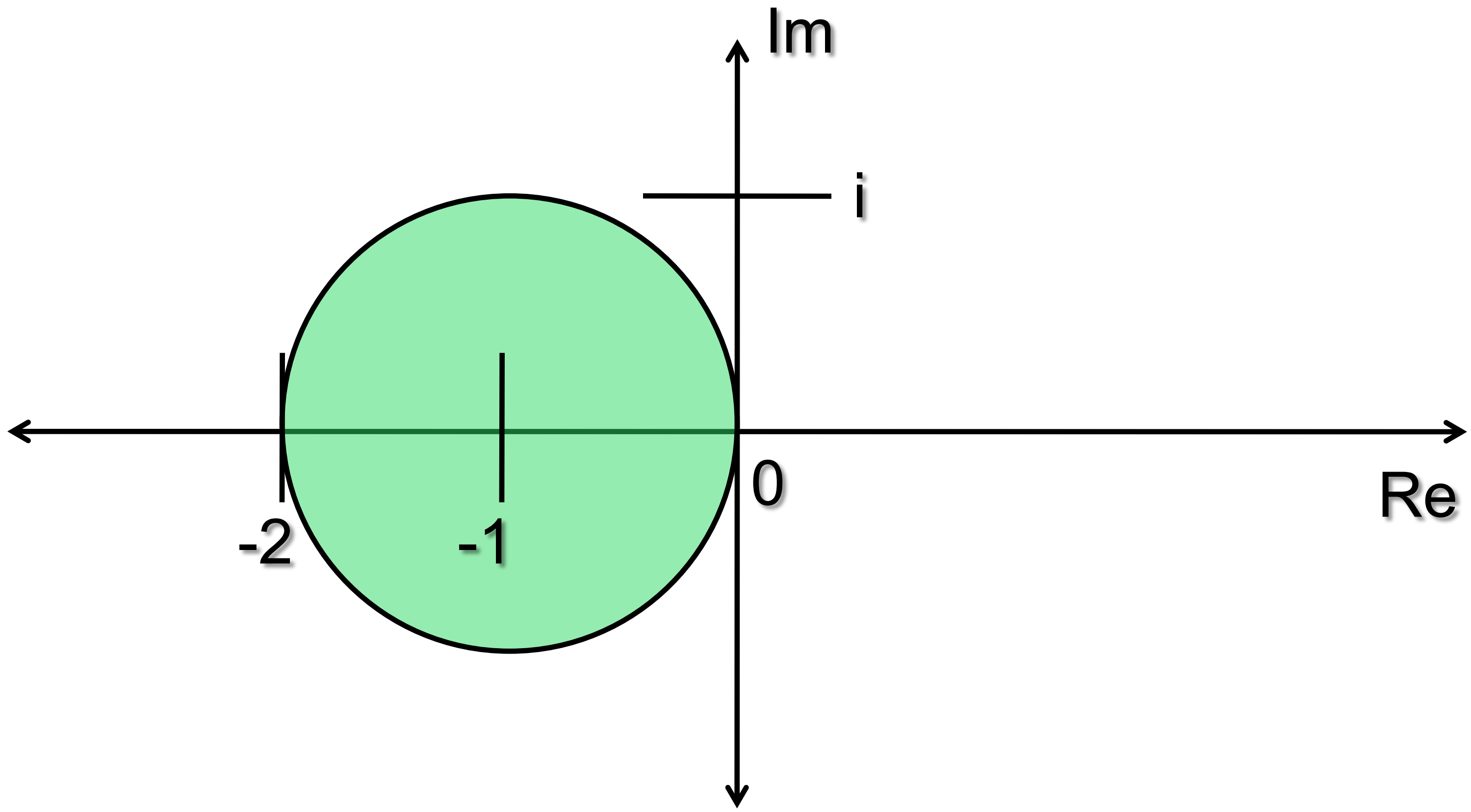
- Compare to semi-discrete method with spatial mesh size Δx:

$$y_i'(t) = \frac{c}{(\Delta x)^2}\left(y_{i+1}(t) - 2y_i(t) + y_{i-1}(t)\right), \quad i = 1, \ldots, n$$

- Finite difference method is equivalent to solving each y_i using Euler's method with $h = \Delta t$

Recall:
Stability region for Euler's method

- Requires eigenvalues of $h_k J_f$ inside



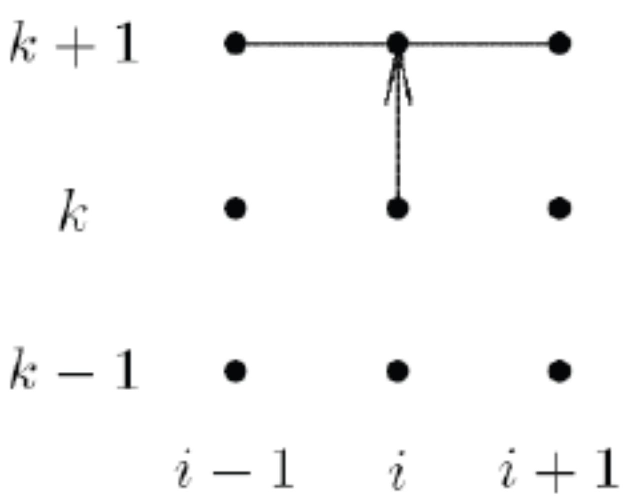
Example, Continued

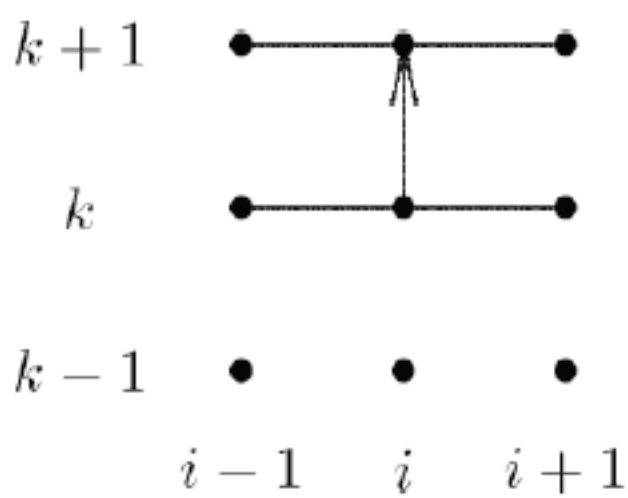
- What is J_f here?

$$y' = \frac{c}{(\Delta x)^2} \begin{bmatrix} -2 & 1 & 0 & \cdots & 0 \\ 1 & -2 & 1 & \cdots & 0 \\ 0 & 1 & -2 & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & 1 & -2 \end{bmatrix} y = Ay$$

- A is Jf, so eigenvalues of $\Delta t A$ must lie inside the circle
- i.e., $\Delta t \leq (\Delta x)^2 / 2c$
- Quite restrictive on Δt!

Alternative Stencils





- Unconditionally stable with respect to Δt
- (Again, no comment on accuracy)

Lax Equivalence Theorem

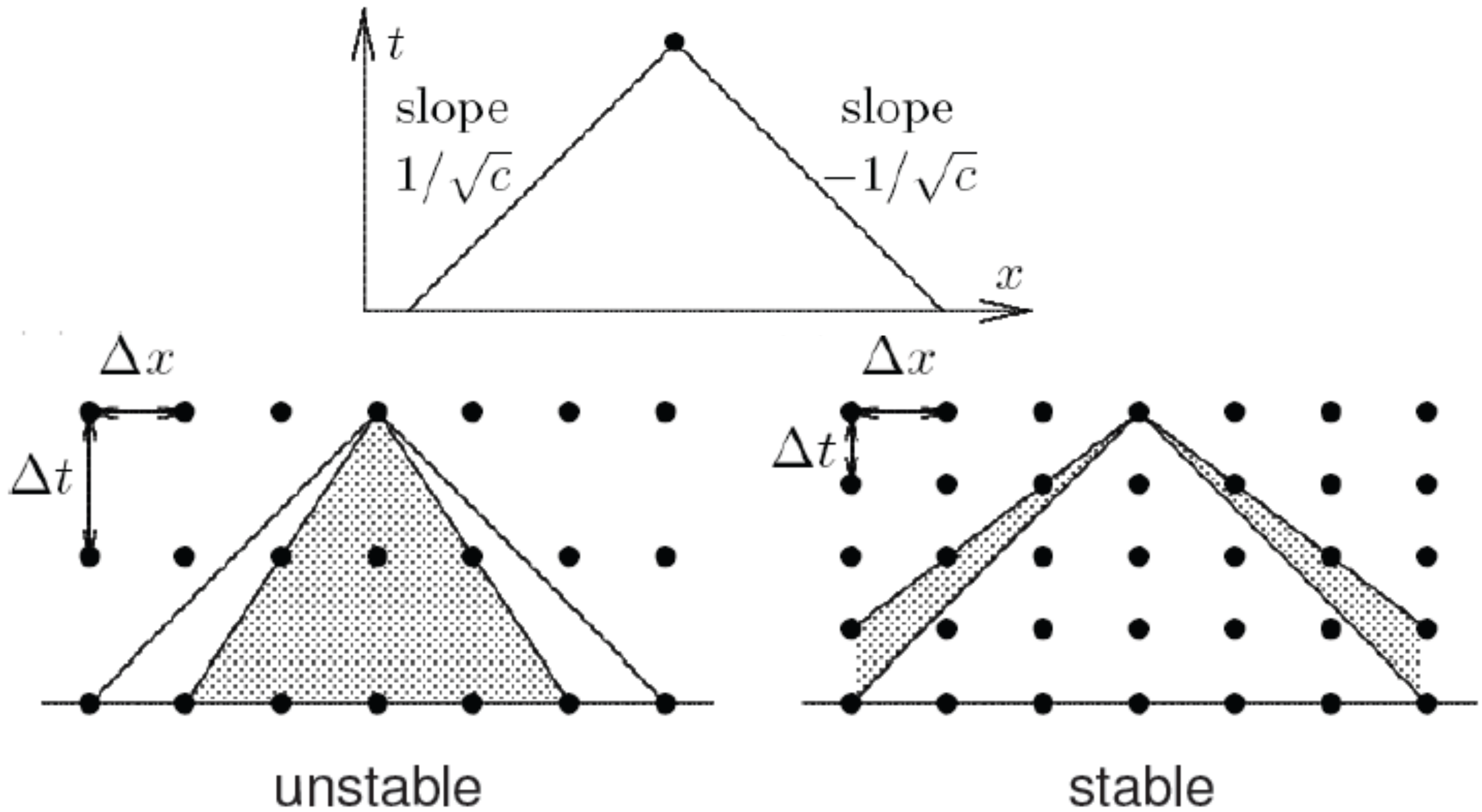
- For a well-posed linear PDE, two necessary and sufficient conditions for finite difference scheme to converge to true solution as Δx and $\Delta t \rightarrow 0$:
  - Consistency: local truncation error goes to zero
  - Stability: solution remains bounded
  - Both are required
- Consistency derived from soundness of approximation to derivatives as $\Delta t \rightarrow 0$
  - i.e., does numerical method approximate the correct PDE?
- Stability: exact analysis often difficult (but less difficult than showing convergence directly)

Reasoning about PDE Stability

- Matrix method
  - Shown on previous slides
- Domains of dependence
- Fourier / Von Neumann stability analysis

Domains of Dependence

- CFL Condition: For each mesh point, the **domain of dependence of the PDE** must lie within the **domain of dependence of the finite difference scheme**



Notes on CFL Conditions

- Encapsulated in “CFL Number” or “Courant number” that relates Δt to Δx for a particular equation
- CFL conditions are necessary but not sufficient
- Can be very restrictive on choice of Δt
- Implicit methods may not require low CFL number for stability, but still may require low number for accuracy

Fourier / Von Neumann Stability Analysis

- Also pertains to finite difference methods for PDEs
- Valid under certain assumptions (linear PDE, periodic boundary conditions), but often good starting point
- Fourier expansion (!) of solution

$$u(x,t) = \sum a_k(n\Delta t)e^{ikj\Delta x}$$

- Assume

$$a_k(n\Delta t) = (\xi_k)^n$$

  - Valid for linear PDEs, otherwise locally valid
  - Will be stable if magnitude of ξ is less than 1: errors decay, not grow, over time

Review of Methods for Large, Sparse Systems

Why the need?

- All BVPs and implicit methods for time-dependent PDEs yield systems of equations
- Finite difference schemes are typically sparse

$$\boldsymbol{y}' = \frac{c}{(\Delta x)^2}\begin{bmatrix} -2 & 1 & 0 & \cdots & 0 \\ 1 & -2 & 1 & \cdots & 0 \\ 0 & 1 & -2 & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & 1 & -2 \end{bmatrix}\boldsymbol{y} = \boldsymbol{A}\boldsymbol{y}$$

Review: Stationary Iterative Methods for Linear Systems

- Can we formulate $g(x)$ such that $x^*=g(x^*)$ when $\mathbf{A}x^* - b = 0$?
- Yes: let $\mathbf{A} = \mathbf{M} - \mathbf{N}$ (for any satisfying $\mathbf{M}, \mathbf{N}$)

  and let $g(x) = \mathbf{G}x + c = \mathbf{M}^{-1}\mathbf{N}x + \mathbf{M}^{-1}b$
- Check: if $x^* = g(x^*) = \mathbf{M}^{-1}\mathbf{N}x^* + \mathbf{M}^{-1}b$ then

$$
\begin{aligned}
\mathbf{A}x^* &= (\mathbf{M} - \mathbf{N})(\mathbf{M}^{-1}\mathbf{N}x^* + \mathbf{M}^{-1}b) \\
&= \mathbf{N}x^* + b + \mathbf{N}(\mathbf{M}^{-1}\mathbf{N}x^* + \mathbf{M}^{-1}b) \\
&= \mathbf{N}x^* + \mathbf{b} - \mathbf{N}x^* \\
&= b
\end{aligned}
$$

So what?

- We have an update equation:
  $x^{(k+1)} = \mathbf{M}^{-1}\mathbf{N}x^k + \mathbf{M}^{-1}b$
- Only requires inverse of M, not A
- We can choose M to be nicely invertible (e.g., diagonal)

Jacobi Method

- Choose M to be the diagonal of A
- Choose N to be M – A = -(L + U)
  - Note that A != LU here
- So, use update equation:

$$x^{(k+1)} = D^{-1} (b - (L + U)x^{k})$$

Jacobi method

- Alternate formulation: Recall we've got

$$
\begin{aligned}
a_{11}x_1 + \; a_{12}x_2 + \cdots + \; a_{1n}x_n &= b_1 \\
a_{21}x_1 + \; a_{22}x_2 + \cdots + \; a_{2n}x_n &= b_2 \\
\vdots \qquad\quad \vdots \qquad\qquad\quad \vdots \quad &\quad \vdots \\
a_{m1}x_1 + a_{m2}x_2 + \cdots + a_{mn}x_n &= b_m
\end{aligned}
$$

- Store all x_i^k
- In each iteration, set

$$x_i^{(k+1)} = \frac{b_i - \sum_{j \neq i} a_{ij} x_j^{(k)}}{a_{ii}}$$

Gauss-Seidel

- Why make a complete pass through components of x using only x_i^k, ignoring the $x_i^{(k+1)}$ we've already computed?

Jacobi: $$x_i^{(k+1)} = \frac{b_i - \sum_{j \neq i} a_{ij} x_j^{(k)}}{a_{ii}}$$

G.S.: $$x_i^{(k+1)} = \frac{b_i - \sum_{j > i} a_{ij} x_j^{(k)} - \sum_{j < i} a_{ij} x_j^{(k+1)}}{a_{ii}}$$

Notes on Gauss-Seidel

- Gauss-Seidel is also a stationary method
  A = M – N where M = D + L, N = -U
- Both G.S. and Jacobi may or may not converge
  - Jacobi: Diagonal dominance is sufficient condition
  - G.S.: Diagonal dominance or symmetric positive definite
- Both can be **very slow to converge**

Successive Over-relaxation (SOR)

- Let $x^{(k+1)} = (1-w)x^{(k)} + w\, x_{GS}^{(k+1)}$
- If w = 1 then update rule is Gauss-Seidel
- If $w < 1$: Under-relaxation
  - Proceed more cautiously: e.g., to make a non-convergent system converge
- If $1 < w < 2$: Over-relaxation
  - Proceed more boldly, e.g. to accelerate convergence of an already-convergent system
- If $w > 2$: Divergence. ☹

Slow Convergence

- All these methods can be very slow
- Can have great initial progress but then slow down
- Tend to reduce high-frequency error rapidly, and low-frequency error slowly
- Demo: http://www.cse.illinois.edu/iem/fft/itrmthds/

Multigrid Methods

See Heath slides

For more info

- http://academicearth.org/lectures/multigrid-methods