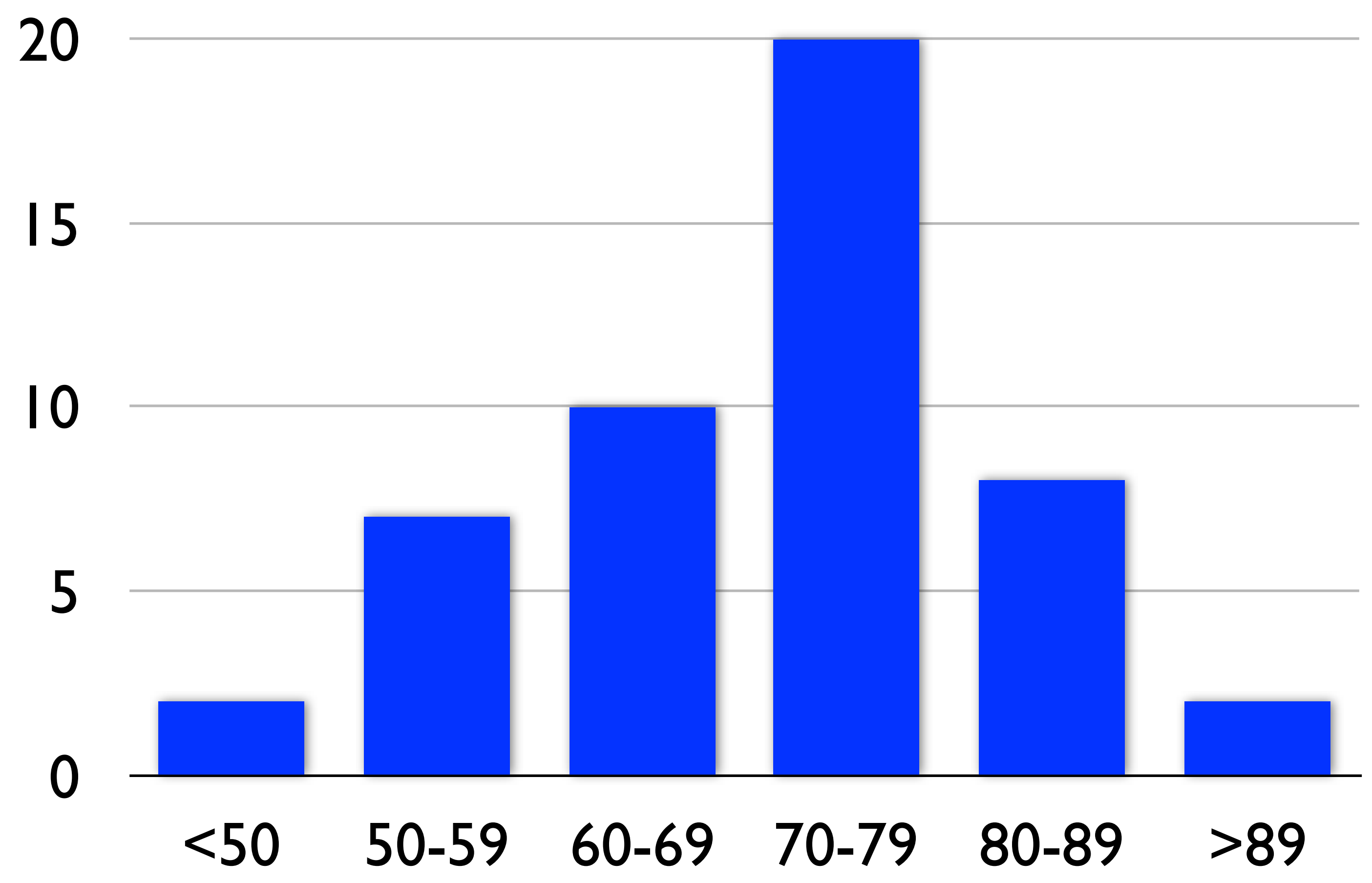
10.34: Numerical Methods Applied to Chemical Engineering

Lecture 16:
ODE-IVP and Numerical Integration

Quiz 1 Results

- Mean: 70.6
- Standard deviation: 11.0



Recap

- Implicit methods for ODE-IVPs

Recap

- Example:
  - Use implicit Euler to solve:

$$\frac{dx}{dt} = \lambda x, x(0) = x_0$$

Give a closed form formula for the numerical solution

Recap

- Example:
  - Use implicit Euler to solve:

$$\frac{dx}{dt} = \lambda x, x(0) = x_0$$

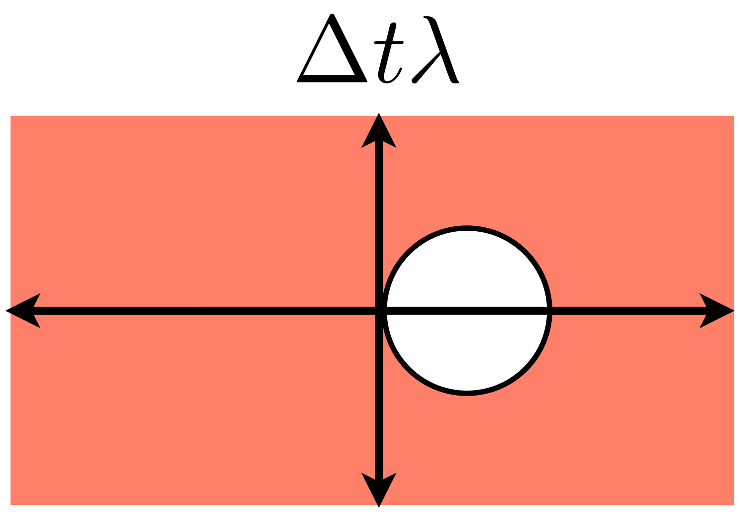
  - Let:

$$x_k = x(k\Delta t)$$

$$x_{k+1} = x_k + \Delta t \lambda x_{k+1}$$

$$x_{k+1} = \frac{1}{1 - \Delta t\lambda} x_k$$

$$x_k = \left(\frac{1}{1 - \Delta t\lambda}\right)^k x_0$$



  - Stability:

$$|1 - \Delta t\lambda| \geq 1 \Rightarrow (1 - \Delta t \mathrm{Re}\lambda)^2 + (\Delta t \mathrm{Im}\lambda)^2 \geq 1$$

Recap

- Example:
  - Use implicit Euler to solve:

$$\frac{dx}{dt} = \lambda x, x(0) = x_0$$

  - Numerical solution:

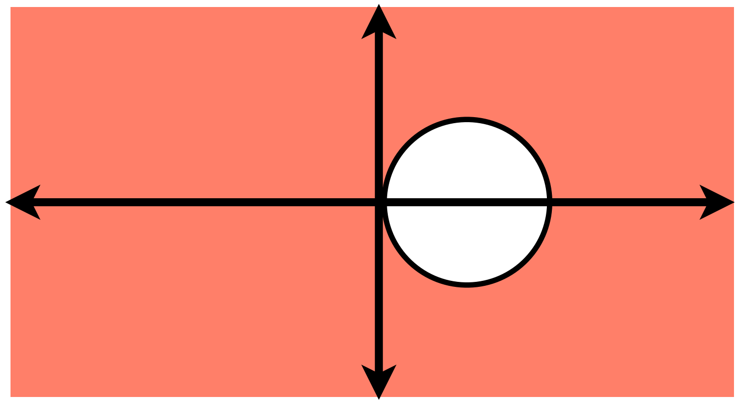
$$x_k = \left(\frac{1}{1 - \Delta t \lambda}\right)^k x_0$$

  - Exact solution:

$$x_k = x_0 e^{k \lambda \Delta t}$$

  - Stability and accuracy do not correlate!

$\Delta t \lambda$



Multistep Methods

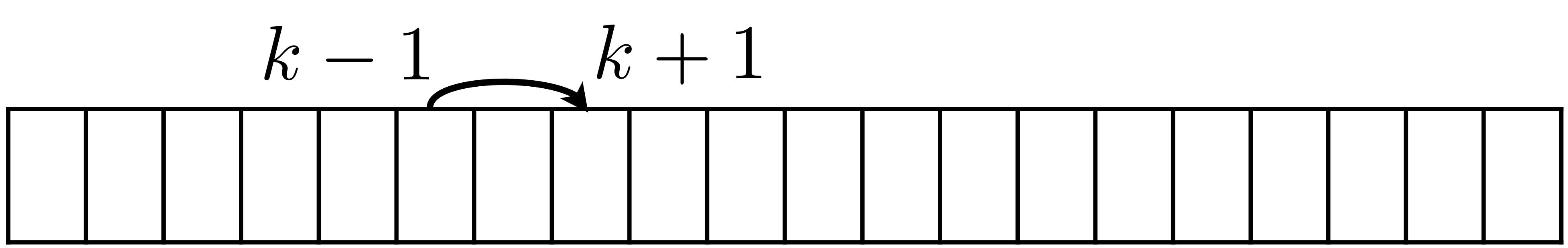
- Multistep methods utilize information over multiple time steps to approximate the solution of an ODE.
- These can be designed for higher accuracy, larger stability bounds or both.
- Example: Leapfrog method

$$\frac{d\mathbf{x}}{dt} = \mathbf{f}(\mathbf{x}(t), t)$$

- Approximate derivative with central difference:

$$\frac{1}{2\Delta t}\left(\mathbf{x}(t+\Delta t) - \mathbf{x}(t-\Delta t)\right) = \mathbf{f}(\mathbf{x}(t), t)$$

$$\mathbf{x}(t_{k+1}) = \mathbf{x}(t_{k-1}) + 2\Delta t \mathbf{f}(\mathbf{x}(t_k), t_k)$$



Multistep Methods

- Local accuracy of the leap frog method:

$$\frac{d\mathbf{x}}{dt} = \frac{1}{2\Delta t}\left(\mathbf{x}(t_{k+1}) - \mathbf{x}(t_{k-1})\right) + O((\Delta t)^2) = \mathbf{f}(\mathbf{x}(t_k), t_k)$$

$$\mathbf{x}(t_{k+1}) = \mathbf{x}(t_{k-1}) + 2\Delta t \mathbf{f}(\mathbf{x}(t_k), t_k) + O((\Delta t)^3)$$

- Stability of the leap frog method: $\frac{dx}{dt} = \lambda x$

$$x_{k+1} = x_{k-1} + 2\Delta t \lambda x_k$$

$$\begin{pmatrix} x_{k+1} \\ x_k \end{pmatrix} = \begin{pmatrix} 2\Delta t\lambda & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} x_k \\ x_{k-1} \end{pmatrix}$$

$$\begin{pmatrix} x_{k+1} \\ x_k \end{pmatrix} = \mathbf{C}^k \begin{pmatrix} x_1 \\ x_0 \end{pmatrix}$$

Multistep Methods

- Local accuracy of the leap frog method:

$$\frac{d\mathbf{x}}{dt} = \frac{1}{2\Delta t}\left(\mathbf{x}(t_{k+1}) - \mathbf{x}(t_{k-1})\right) + O((\Delta t)^2) = \mathbf{f}(\mathbf{x}(t_k), t_k)$$

$$\mathbf{x}(t_{k+1}) = \mathbf{x}(t_{k-1}) + 2\Delta t \mathbf{f}(\mathbf{x}(t_k), t_k) + O((\Delta t)^3)$$

- Stability of the leap frog method: $$\frac{dx}{dt} = \lambda x$$

$$x_{k+1} = x_{k-1} + 2\Delta t \lambda x_k$$

$$\begin{pmatrix} x_{k+1} \\ x_k \end{pmatrix} = \mathbf{C}^k \begin{pmatrix} x_1 \\ x_0 \end{pmatrix}$$

$$\mathbf{C} = \begin{pmatrix} 2\Delta t \lambda & 1 \\ 1 & 0 \end{pmatrix}$$

What are the eigenvalues of this matrix?

Multistep Methods

- Local accuracy of the leap frog method:

$$\frac{d\mathbf{x}}{dt} = \frac{1}{2\Delta t}\left(\mathbf{x}(t_{k+1}) - \mathbf{x}(t_{k-1})\right) + O((\Delta t)^2) = \mathbf{f}(\mathbf{x}(t_k), t_k)$$

$$\mathbf{x}(t_{k+1}) = \mathbf{x}(t_{k-1}) + 2\Delta t\mathbf{f}(\mathbf{x}(t_k), t_k) + O((\Delta t)^3)$$

- Stability of the leap frog method: $\frac{dx}{dt} = \lambda x$

$$x_{k+1} = x_{k-1} + 2\Delta t\lambda x_k$$

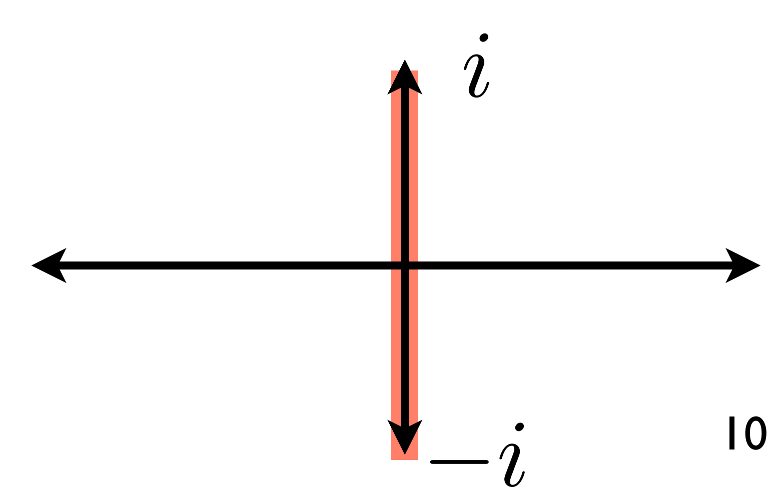
- Both eigenvalues of $\mathbf{C}$ must be bounded:

$$|\Delta t\lambda \pm \sqrt{(\Delta t\lambda)^2 + 1}| \leq 1$$

consider when $|\Delta t\lambda| \ll 1$:

$$|\Delta t\lambda \pm 1| \leq 1$$

$\Delta t\lambda$



Multistep Methods

- Exercise:
  - Should I use the leap frog method to integrate the equations of motion for a mass-spring system?

$$m\frac{d^2x}{dt^2} = -kx$$

    - If so, what time steps should I limit myself to?
    - If not, what other integrator could I use?

Multistep Methods

- Exercise:
  - Should I use the leap frog method to integrate the equations of motion for a mass-spring system?

$$m\frac{d^2x}{dt^2} = -kx$$

  - Transform to system of first order ODEs:

$$\frac{d}{dt}\begin{pmatrix} v \\ x \end{pmatrix} = \begin{pmatrix} 0 & -k/m \\ 1 & 0 \end{pmatrix}\begin{pmatrix} v \\ x \end{pmatrix}$$

  - Eigenvalues of matrix: $\lambda = \pm i\sqrt{\frac{k}{m}}$
  - Since eigenvalues are imaginary, leap frog is stable when:

$$\Delta t < \sqrt{\frac{m}{k}}$$

Multistep Methods

- Multistep methods can be implicit as well such as the backward differentiation formulas or Adams-Moulton integrators.
- Example: Backwards differentiation

$$\mathbf{x}_{k+1} = \frac{4}{3}\mathbf{x}_k - \frac{1}{3}\mathbf{x}_{k-1} + \frac{2}{3}\Delta t \mathbf{f}(\mathbf{x}_{k+1}, t_{k+1})$$

  - Second order accurate.
  - How would you identify the stability bounds?

Numerical Integration

- Consider the definite integral:

$$\int_{t_0}^{t_f} \mathbf{f}(\tau)d\tau$$

- We can define a variable:

$$\mathbf{x}(t) = \int_{t_0}^{t} \mathbf{f}(\tau)d\tau$$

  - which, if $\mathbf{f}(t)$ is continuous, satisfies the differential equation:

$$\frac{d}{dt}\mathbf{x}(t) = \mathbf{f}(\tau), \quad \mathbf{x}(t_0) = 0$$

- Thus, a definite integral of a known, continuous function can be determined using methods for ODE-IVPs to compute:

$$\mathbf{x}(t_f)$$

Numerical Integration

- Consider the definite integral:

$$\int_{t_0}^{t_f} \mathbf{f}(\tau) d\tau$$

- If the discontinuities in $\mathbf{f}(t)$ are known, then ODE-IVP solvers can be used in the domain between the discontinuities too!

- If the discontinuities in $\mathbf{f}(t)$ are unknown, then Monte-Carlo methods (discussed later are a better option).

- This approach is efficient with adaptive time stepping methods because an appropriate spacing between points can be chosen when t changes more or less rapidly with $\mathbf{f}(t)$

- For multi-dimensional integrals, this approach is not as straightforward, however.

Numerical Integration

- One alternative is integration by polynomial interpolation:

$$\int_{t_0}^{t_f} \mathbf{f}(\tau)d\tau = \sum_{k=1}^{N} \int_{t_{k-1}}^{t_k} \mathbf{f}(\tau)d\tau \approx \sum_{k=1}^{N} \int_{t_{k-1}}^{t_k} \mathbf{P}_k(\tau)d\tau$$

  - where $\mathbf{P}_k(\tau)$ is a polynomial approximation of $\mathbf{f}(\tau)$ in the domain

$$\tau \in [t_{k-1}, t_k]$$

  - If the size of the domains of integration and the order of the polynomial interpolant can be used to control the accuracy of the integration.

  - Example: quadratic interpolation – Simpson's rule:

$$\mathbf{P}_k(\tau) = \mathbf{f}(t_{k-1}) + \frac{1}{t_k - t_{k-1}} \left(\mathbf{f}(t_k) - \mathbf{f}(t_{k-1})\right)(\tau - t_{k-1})$$

$$\int_{t_{k-1}}^{t_k} \mathbf{P}_k(\tau)d\tau = \frac{1}{2} \left(\mathbf{f}(t_k) + \mathbf{f}(t_{k-1})\right)(t_k - t_{k-1})$$

Numerical Integration

- One alternative is integration by polynomial interpolation:

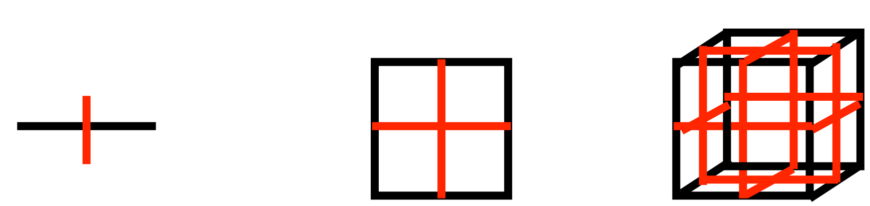
$$\int_{t_0}^{t_f} \mathbf{f}(\tau)d\tau = \sum_{k=1}^{N} \int_{t_{k-1}}^{t_k} \mathbf{f}(\tau)d\tau \approx \sum_{k=1}^{N} \int_{t_{k-1}}^{t_k} \mathbf{P}_k(\tau)d\tau$$

  - where $\mathbf{P}_k(\tau)$ is a polynomial approximation of $\mathbf{f}(\tau)$ in the domain $\tau \in [t_{k-1}, t_k]$
  - If the size of the domains of integration and the order of the polynomial interpolant can be used to control the accuracy of the integration.
  - Example: quadratic interpolation – Simpson's rule:

$$\int_{t_{k-1}}^{t_k} \mathbf{P}_k(\tau)d\tau = \frac{1}{6}\left(\mathbf{f}(t_k) + 4\mathbf{f}((t_k + t_{k-1})/2) + \mathbf{f}(t_{k-1})\right)(t_k - t_{k-1})$$

Numerical Integration

- Multidimensional integration:
  - Of the sort: $$\int_{y_L}^{y_U} \int_{z_L}^{z_U} \mathbf{f}(y,z) dy dz$$
    - For any number of dimensions larger than 3, this is best handled with Monte Carlo methods
    - For dimensions less than 3, this integration can be done with polynomial interpolation.
      - Fit the function to a polynomial of a prescribed degree within small regions of the domain of integration.
      - Sum integrals over the polynomial fits in each fit region.
      - This fails with higher dimensions because the number of fit regions grows exponentially with dimension.
        - Example:



Numerical Integration

- Improper integrals:
  - Of the sort: $$\int_{t_0}^{\infty} \mathbf{f}(\tau) d\tau$$
  - Can be split into two domains of integration

$$\int_{t_0}^{\infty} \mathbf{f}(\tau) d\tau = \int_{t_0}^{t_f} \mathbf{f}(\tau) d\tau + \int_{t_f}^{\infty} \mathbf{f}(\tau) d\tau$$

    - The first integral can be handled with ODE-IVP methods or polynomial interpolation
    - The second must be handled separately through either:
      - transformation onto a finite domain
      - or substitution of an asymptotic approximation
  - This same idea applies to integrable singularities as well.

Numerical Integration

- Improper integrals:
  - Example:

$$\int_0^{t_f} \frac{\cos\tau}{\sqrt{\tau}} d\tau$$

$$\approx \int_0^{t_0} \frac{1-\tau^2/2}{\sqrt{\tau}} d\tau + \int_{t_0}^{t_f} \frac{\cos\tau}{\sqrt{\tau}} d\tau$$

$$\approx 2t_0^{1/2} - \frac{1}{5} t_0^{5/2} + \int_{t_0}^{t_f} \frac{\cos\tau}{\sqrt{\tau}} d\tau$$