Outline

Introduction
- Numerical methods for ODEs
- Accuracy
- Stability
- Applications
- Preserving invariants
- Preserving symmetry
- Preserving geometric structures

Some Challenges
- Stiff ODEs
- Constrained dynamics
- Coarse-graining
- Multiple time scales
- Statistical mechanics
Ordinary Differential Equations

- General form: \( x' = f(x, t) \)
- Applications
  - Mechanics
  - Molecular models
  - Chemical reactions
  - Discretization of PDEs
- Examples from 251
  - Mass-spring model, population model, motion in space ...
- For most ODEs, the solutions can be obtained from analytical methods.
Numerical Solutions

- Exact solution \( x(t) = \Phi(t, x(0)) \)
- Numerical solution \( x(t) = \Psi(t, x(0); \Delta t) \)
- Numerical methods
  - Time discretization \( \tau = \{t_0, t_1, \cdots, t_N\} \subset [0, T] \)
  - Example: uniform step size: \( t_j = j\Delta t, \ j \geq 0 \)
  - Euler’s method:

\[
\frac{x(t_{j+1}) - x(t_j)}{\Delta t} = f(x(t_j), t_j)
\]
A Two-Stage Runge-Kutta Method

✧ First stage: \[ k_1 = f(x(t_j), t_j), \quad x(t_{j+1})^* = x(t_j) + \Delta t k_1 \]

✧ Second stage:

\[ k_2 = f(x(t_{j+1})^*, t_{j+1}), \quad x(t_{j+1}) = x(t_j) + \frac{\Delta t}{2} [k_1 + k_2] \]

✧ Numerical error

\[ \| \Phi(t) - \Psi(t) \| \leq \Delta t^2 e^{Lt} \]

✧ Higher order RK methods are available (try rk45 in Matlab)
Inheritance of Asymptotic Stability

- For a linear system, \( x' = Ax, \quad x = 0 \) is stable if all eigenvalues of \( A \) are negative.

- For a nonlinear system, \( x' = f(x) \), an equilibrium \( x_0 \) is stable if the eigenvalues of \( \nabla f(x_0) \) are negative.

- A numerical method is stable if the stability of the linear system is inherited.

- Typically, the step size has to be sufficiently small (inverse proportional to the eigenvalues) in order for the method to be stable.

- The problem becomes \textit{stiff} when some eigenvalues are large.
Many-Particle Models

- Coordinate and Momentum \((q_i, p_i), i = 1, 2, \ldots, N\).
- The equations:
  \[ \dot{q}_i = \frac{p_i}{m_i} \]
  \[ \dot{p}_i = -\sum_{j \neq i} V'(|q_i - q_j|) \frac{q_i - q_j}{|q_i - q_j|} \]
- Linear momentum:
  \[ P = \sum_i p_i \]
- Angular momentum:
  \[ L = \sum_i q_i \times p_i \]
- Total energy:
  \[ H = \sum_i \frac{p_i^2}{2m_i} + \frac{1}{2} \sum_{i \neq j} V(|q_i - q_j|) \]
- Conservation:
  \[ \frac{d}{dt} P = 0, \quad \frac{d}{dt} L = 0, \quad \frac{d}{dt} H = 0. \]
First Integrals (Invariants)

- In general, for \( y' = f(y) \)
- \( I(y) \) is a first integral if \( \frac{d}{dt} I(y) = 0 \).
- Most numerical methods preserve linear invariant
- Only some methods preserve quadratic invariants
- In general, it is not possible to exactly preserve invariants of higher order.
Symmetry

- If \( \rho f(y) = -f(\rho y) \) the function is symmetric \( \text{wrt} \) \( \rho \).

- The solutions of the ODEs will be called \( \rho \) - reversible.

- \( \rho \circ \Phi = \Phi^{-1} \circ \rho \)

- For example, \( (q_i, p_i) \rightarrow (q_i, -p_i) \)

- A numerical method is \( \rho \) - reversible if the solution satisfies the same properties.

- Explicit RK methods are not symmetric.
Symplectic Structure

\[ J = \begin{bmatrix} 0 & I \\ -I & 0 \end{bmatrix} \]

- A linear mapping \( A \) is symplectic if \( A^T J A = J \)
- A nonlinear mapping \( g \) is symplectic if
  \[ \nabla g(q, p)^T J \nabla g(q, p) = J \]
Hamiltonian systems

- Hamilton's principle
  \[ \dot{q} = \frac{\partial H}{\partial p} \]
  \[ \dot{p} = -\frac{\partial H}{\partial q} \]

- Examples: mass spring model, many-body problems, pendulum model, Lotka-Volterra model etc.

- The mapping \((p(0), q(0)) \rightarrow (p(t), q(t))\) is symplectic.

- A numerical method is symplectic if it defines a symplectic mapping.
Symmetric and symplectic methods

❖ Symmetric methods
  ❖ Order of accuracy is always even
  ❖ Very convenient for an extrapolation procedure

❖ Symplectic methods
  ❖ Very good energy conservation properties
  ❖ Promising accuracy over long time integration
  ❖ Provide good statistics.
Dimension Reduction: I

- A Hamiltonian system \( y' = J\nabla H(y) \)
- Imposing constraints \( c(y) = 0, c : \mathbb{R}^N \rightarrow \mathbb{R}^{N-n} \)
- Effectively, there are only \( n \) free variables
- Example: flexible pendulum

\[
H = K + U = \frac{1}{2}(mv_1^2 + mv_2^2) + mgx_2 + \frac{1}{2\epsilon^2}\left(\sqrt{x_1^2 + x_2^2} - l\right)^2
\]
Dimension Reduction: II

- A large system \( x' = -Ax, x \in \mathbb{R}^N, A \in \mathbb{R}^{N \times N}, N \gg 1 \).

- Some quantities of interest \( y = Bx, y \in \mathbb{R}^n, B \in \mathbb{R}^{n \times N} \).

- Remaining degrees of freedom:
  \[ z =Cx, \quad z \in \mathbb{R}^{N-n}, B \in \mathbb{R}^{(N-n) \times N} \]

- Effective equation:
  \[ M y' = -Ky + \int_0^t S(t - s)y(s)ds + R(t) \]
  - \( S(t) \): memory function
  - \( R(t) \): random noise
Summer Project I

- Minimum energy path (MEP)
- The most efficient path from one stable state to another
- The most probable path
- Represents rare but important events
- ODEs with boundary conditions
Summer Project II

- ODEs with multiple time scales
- An example,
  \[
  \begin{cases}
    x' &= f(x, y) \\
    y' &= -\frac{1}{\varepsilon}(y - \varphi(x))
  \end{cases}
  \]
- A multiscale method targeting the slow variables only
- Large time step size