MATH 3341: Introduction to Scientific Computing Lab

Libao Jin

University of Wyoming

December 02, 2020

Lab 14: Built-in ODE Solvers in MATLAB

Built-in ODE Solvers for Stiff/Nonstiff ODEs

Stiff ODEs

Definition

A stiff equation is a differential equation for which certain numerical methods for solving the equation are numerically unstable, unless the step size is taken to be extremely small. It has proven difficult to formulate a precise definition of stiffness, but the main idea is that the equation includes some terms that can lead to rapid variation in the solution.

Choose an ODE Solver

- Nonstiff ODE Solvers: `ode45`, `ode23`, and `ode113`
- Stiff ODE Solvers: `ode15s`, `ode23s`, `ode23t`, and `ode23tb`
- Fully Implicit ODE Solvers: `ode15i`

Choose an ODE Solver

Some ODE problems exhibit *stiffiness*, or difficulty in evaluation. For example, if an ODE has two solution components that vary on drastically different time scales, then the equation might be stiff. You can identify a problem as stiff if nonstiff solvers (such as `ode45`) are unable to solve the problem or are extremly slow. If you observe that a nonstiff solver is very slow, try using a stiff solver such as `ode15s` instead. When using a stiff solver, you can improve reliability and efficiency by supplying the Jacobian matrix or its sparsity pattern.

ode45: Solve non-stiff ODEs, medium order method

- `[t, y] = ode45(f, [t0 tfinal], y0)` integrates the system of differential equations $y' = f(t, y)$ from time `t0` to `tfinal` with initial conditions `y0`. `f` is a function handle.
- `[t, y] = ode45(f, t_span, y0)` with `t_span = [t0, t1, t2, ..., tfinal]` integrates the system of differential equations $y' = f(t, y)$ from time `t0` to `tfinal` with initial conditions `y0`. `f` is a function handle. In this case, `t` is same as `t_span`.
- Example: Solving a separable ODE $y' = 4t$ with $y_0 = 0$.

$$y' = \frac{dy}{dt} = 4t \implies dy = 4t\,dt \implies \int 1\,dy = \int 4t\,dt.$$

Therefore, $y = 2t^2 + C$ and
$y_0 = y(0) = 0 \implies C = 0 \implies y(t) = 2t^2$. In MATLAB:

```
f = @(t, y) 4 * t; y0 = 0; t_span = linspace(0, 5, 50);
[t, y] = ode45(f, t_span, y0);
```

ode45: Solving higher order ODEs

- Convert the higher order ODEs into a system of first-order ODEs, then solve it using `ode45`.
- Example: $y'' - (2 - y^2)y' + y = 0$ with $y(1) = 2, y'(1) = 0$.
  Let $y_1 = y$ and $y_2 = y'$, rewriting the second order ODE gives

$$y'' = (2 - y^2)y' - y \implies \begin{cases} y_1' = y' = y_2 \\ y_2' = y'' = (2 - y_1^2)y_2 - y_1 \end{cases}$$

In matrix form, we have

$$\mathbf{y}' = \begin{bmatrix} y_1 \\ y_2 \end{bmatrix}' = \begin{bmatrix} y_1' \\ y_2' \end{bmatrix} = \begin{bmatrix} y_2 \\ (2 - y_1^2)y_2 - y_1 \end{bmatrix}.$$

In MATLAB:

```
f = @(t, y) [y(2); (2 - y(1)^2) * y(2) - y(1)];
t_span = linspace(1, 3, 100); y0 = [2; 0];
[t,y] = ode45(f,t_span,y0)  % y(:,1) is the solution
```