Differential Equations

September 28, 2023

Ordinary Differential Equation

- A smooth function $y(t)$ governed by $y'(t) = f(t, y(t)), t \in [0, T]$
- $f(t, y(t))$ is continuous and satisfies Lipschitz condition wrt y,
- $y(t)$ has a unique solution for $0 \leq t \leq T$, given initial value $y(0) = y_0$.
- Numerical solutions, generate approximation u_i of $y(t_i)$ over a sequence of time steps, $t_i, i = 0, \cdots, N$ with $u_0 = y_0$.

Ordinary Differential Equation

Explicit Methods

- Euler's method:

$$u_{i+1} = u_i + h\phi(t_i, u_i), h = t_{i+1} - t_i$$

  , one step method, global relative error $\mathcal{O}(h)$.

- Runge-Kutta methods: high order accurate 1-step methods, multiple evaluations of $f(t_i + \Delta t, u_i + k)$ for some $\Delta t \in [0, h]$ and some shock amount k, infeasible in practice when $f(t, y(t))$ is availabe only on exact grid points t_i.

Ordinary Differential Equation

Explicit Methods (continued)

- Multiple step methods:

$$\begin{aligned} u_{i+1} &= \alpha_m u_i + \alpha_{m-1} u_{i-1} + \cdots + \\ &\quad h\left[\beta_m f(t_i, u_i) + \beta_{m-1}(t_{i-1}, u_{i-i}) + \cdots + \beta_0 f(t_{i-m}, u_{i-m})\right] \end{aligned}$$

- Examples: Adams-Bashforth methods, Adams-Moulton methods.
- $f(t, y(t))$ is evaluated exactly on grid points and computed only once at each t_i.
- Stability issue when h is not fine enough.
- Excellent choice for smooth problems.

Ordinary Differential Equation

Classic Iterative Deferred Correction (IDeC)

- Consider using a base ODE method to obtain d-th order accurate approximation $u^{(d)}(t_i)$ of $y(t_i)$.
- Construct a smooth interpolation function $L(t)$ passing through $p+1$ number of points $(u_{i-p}^{(d)}, \cdots, u_i^{(d)})$.
- The error function $e(t) = y(t) - L(t)$ then satisfies the ODE

$$e'(t) = f(t, y(t)) - L'(t)$$

- We again solve $e(t)$ using Euler's method

$$e_i = e_{i-1} + h\left[f(t_{i-1}, u_{i-1}^{(d)} + e_{i-1}) - L'(t_{i-1})\right]$$

Ordinary Differential Equation

Classic IDeC (continued)

- Adding e_i to $u_i^{(d)}$ and obtain $\mathcal{O}(h^{d+1})$ accurate $u_i^{(d+1)} = e_i + u_i^{(d)}$
- Repeat q number error correction iterations to raise the accuracy order to $\mathcal{O}(h^{d+q})$.
- Mathwrist NPL uses Lagrange interpolation polynomial $L(t)$ and restricts $3 < p < 7$.
- The number of IDeC iterations $q < p$. Extra iterations no longer increase accuracy.
- Mathwrist NPL uses Euler's method or Adams-Bashforth 3-step method as the base method.

Ordinary Differential Equation

Spectral IDec Method

- Rewrite ODE initial value problem in an integral equation form,

$$y(t) = y_0 + \int_0^t f(s, y(s))ds$$

- Obtain d-th order accurate approximation $u^{(d)}(t)$ from a base method.
- Define a residual function:

$$\epsilon(t) = y_0 + \int_0^t f(s, u^{(d)}(s))ds - u^{(d)}(t)$$

Ordinary Differential Equation

Spectral IDec Method (continued)

- Error function $e(t) = y(t) - u^{(d)}$ then can be expressed as a Picard integral equation

$$\begin{aligned} e(t) &= \int_0^t g(s, e(s))ds + \epsilon(t) \\ g(t, e(t)) &= f(t, u^{(d)}(t) + e(t)) - f(t, u^{(d)}(t)) \end{aligned}$$

- Solve $e(t)$ i.e. from explicit Euler's method as

$$e_i = e_{i-1} + g(t_{i-1}, e_{i-1})\Delta t_i + \epsilon_i - \epsilon_{i-1}$$

  , where ϵ_i is computed from Gaussian quadrature.

- Mathwrist NPL uses a Legendre polynomial of degree p to compute the quadrature, which has $2p - 2$ order accuracy on $\epsilon(t)$.

Ordinary Differential Equation

Implicit IDec Method

- Explicit ODE methods need use small enough step size h to retain numerical stability.
- For stiff problems, h could be extremely small.
- A n-dimensional system of ODE is linear if

$$y'(t) = \mathbf{A}(t)y(t) + s(t)$$

, where $\mathbf{A}(t)$ is a $n \times n$ matrix. If the source function $s(t) = 0$, the system is homogeneous.

- The system is stable if all eigen values of $\mathbf{A}(t)$ are negative.

Ordinary Differential Equation

Implicit IDec Method (continued)

- Using implicit Euler's method as the base method i.e. in a classic IDeC,

$$\mathbf{y}_{i+1} = \mathbf{y}_i + h\left[\mathbf{A}_{i+1}\mathbf{y}_{i+1} + \mathbf{s}_{i+1}\right]$$

- This is to solve a linear system

$$\left[\mathbf{I} - h\mathbf{A}_{i+1}\right]\mathbf{y}_{i+1} = \mathbf{y}_i + h\mathbf{s}_{i+1}$$

- Since all eigen values of $\mathbf{A}(t)$ are negative, the implicit Euler is unconditionally stable, hence renders relatively large step size h.
- Mathwrist NPL offers implicit Euler's method combined with classic IDeC or spectral IDeC.

Parabolic Partial Differential Equation

- A 2-d smooth function $v(t, x)$ in a rectangle area $(t, x) \in (a, b) \times (0, T)$ governed by parabolic PDE,

$$\frac{\partial v}{\partial t} = f(t, x)\frac{\partial^2 v}{\partial x^2} + g(t, x)\frac{\partial v}{\partial x} + h(t, x)v + s(t, x) \tag{1}$$

- Conveniently, introduce spatial derivative operator $\mathcal{A}(v)$ and express the PDE in an ODE form,

$$\frac{\partial v}{\partial t} = \mathcal{A}(v) + s(t, x) \tag{2}$$

- The PDE is linear because coefficient functions $f(t, x)$, $g(t, x)$ and $h(t, x)$ do NOT dependent on $v(t, x)$.
- It is homogeneous when the source function $s(t, x) = 0$.
- For a well-posed problem, $f(t, x) > 0$.

Parabolic Partial Differential Equation

Initial Value and Boundary Condition

- Given initial value $v(0,x) = u(x)$, PDE (1) has a unique solution subject to appropriate boundary conditions:
  1. Dirichlet:
  $$v(t, x_b) = u_{bc}(t)$$
  2. Neumann:
  $$\frac{\partial v}{\partial x}(t, x_b) = u_{bc}(t)$$
  3. Robin: a linear combination of Dirichlet and Neumann.
  $$\alpha v(t, x_b) + \beta \frac{\partial v}{\partial x}(t, x_b) = u_{bc}(t)$$

  , where x_b is the boundary state a or b for some given function $u_{bc}(t)$.
- Other boundary conditions lead to no solution.

Parabolic Partial Differential Equation

Finite Difference Method

- Consider a mesh $\{t_l, l = 0, \cdots, N\} \times \{x_j, j = 0, \cdots, J\}$ that is uniformly discretized in the spatial dimension,
- Denote function values at (t_l, x_j) with subscript (l, j), e.g. $f_{l,j} = f(t_l, x_j)$,
- Approximate $\frac{\partial v}{\partial t}$, $\frac{\partial v}{\partial x}$ and $\frac{\partial^2 v}{\partial x^2}$ by finite difference,
- At mesh grid points, PDE (1) becomes to a difference equation in terms of nodal values $v_{l,j}$ that approximates $v(t_l, x_j)$.
- Numerical solution is obtained by sloving all nodal values $v_{l,j}$ simultaneously to satisfy the difference equation.

Parabolic Partial Differential Equation

Crank-Nicolson Method

- Crank-Nicolson uses 3-point center-difference scheme in spatial dimension,

$$\left.\frac{\partial v}{\partial x}\right|_{t=t_l,x=x_j} = \frac{v_{l,j+1} - v_{l,j-1}}{2\Delta x}$$

$$\left.\frac{\partial^2}{\partial x^2}\right|_{t=t_l,x=x_j} = \frac{v_{l,j+1} - 2v_{l,j} + v_{l,j-1}}{\Delta x^2}$$

- Along time dimension,

$$\left.\frac{\partial v}{\partial t}\right|_{t=t_l,x=x_j} = \frac{v_{l+1,j} - v_{l,j}}{\Delta t} = \frac{1}{2}\left(w_{l+1,j} + s_{l+1,j}\right) + \frac{1}{2}\left(w_{l,j} + s_{l,j}\right)$$

$$w_{l,j} = f_{l,j}\left.\frac{\partial^2}{\partial x^2}\right|_{t=t_l,x=x_j} + g_{l,j}\left.\frac{\partial v}{\partial x}\right|_{t=t_l,x=x_j} + h_{l,j}$$

Parabolic Partial Differential Equation

Crank-Nicolson (continued)

- PDE (1) becomes to

$$\frac{v_{l+1,j}}{\Delta t} - \frac{1}{2} L_{l+1,j} = \frac{v_{l,j}}{\Delta t} + \frac{1}{2} L_{l,j} + \frac{1}{2} \left[s_{l+1,j} + s_{l,j} \right]$$

$$L_{l,j} = f_{l,j} \frac{v_{l,j+1} - 2v_{l,j} + v_{l,j-1}}{\Delta x^2} + g_{l,j} \frac{v_{l,j+1} - v_{l,j-1}}{2\Delta x} + h_{l,j} v_{l,j}$$

- At boundary states x_0 and x_J, spatial derivatives involve "ghost" point x_{-1} and x_{J+1}, imply nodal values at ghost points from boundary conditions, i.e. for Robin boundary condition,

$$v_{l,-1} = \frac{2\alpha_1 \Delta x}{\beta_1} v_{l,0} + v_{l,1} - \frac{2\Delta x}{\beta_1} u_1(t_l)$$

Parabolic Partial Differential Equation

Crank-Nicolson (continued)

- Solve a $(J+1) \times (J+1)$ tridiagonal linear system at each time marching step.

$$\mathbf{B}_{l+1} v_{l+1} = \mathbf{A}_l v_l + \mathbf{s}$$

- Second order accuracy $\mathcal{O}(h^2)$ and $\mathcal{O}(\Delta x^2)$ in both time and states.
- Unconditionally stable.
- Handle non-smooth (even non-differentiable) initial value function $v(0, x) = u(x)$.
- To improve accuracy, Mathwrist NPL allows users to set an initial smoothing time $0 < s < T$. We internally will start with a 4-th order accurate finite difference method to smooth $v(t, x)$ out to time s and then switch back to Crank-Nicolson method.

Parabolic Partial Differential Equation

Spectral Collocation Method

- Approximate $v(t,x)$ by a degree-n approximation polynomial $u(t,x)$ in terms of orthogonal basis polynomials (Mathwrist NPL uses Chebyshev basis polynomials),

$$\begin{aligned} u(t,x) &= \sum_{k=0}^{n} a_k(t) T_k(x), k = 0, \cdots, n \\ T_k(x) &= \cos\left(k \cos^{-1} x\right) \end{aligned}$$

- We then require

$$\left.\frac{\partial}{\partial t} u\right|_{x=x_j} = \mathcal{A}(u) + s(t, x_j) \tag{3}$$

, at some representative node points $x_j \in [a,b]$, known as collocation points.

Parabolic Partial Differential Equation

Spectral Collocation Method (continued)

- For Chebyshev basis polynomials, x_j are choosen to be the extreme points of $T_n(x)$, $x_j = \cos \frac{\pi j}{n}, j = 0, \cdots, n$.
- Collocation equation (3) then is n-dimensional system of linear ODE.
- Mathwrist NPL solves this ODE system using implicit spectral IDeC method.
- When $v(t, x)$ is a very smooth function in states, spectral collocation method outperforms all other methods. It doesn't work well if not so.
- It doesn't work for non-smooth or non-differentiable initial value functions.
- However, we allow users to set an initial smoothing time $0 < s < T$ and use a 4-th order accurate finite difference method to smooth $v(t, x)$ to s.

Method of Lines

- Method of lines refer to the technique of solving PDE (1) by solving a system of ODE (3).
- Spectral collocation method falls into this category.
- Alternatively, we can approximate $\mathcal{A}(u)$ using finite difference.
- Mathwrist NPL provides a 2nd order and a 4th order accurate finite difference method, and then solve (3) by implicit classic IDeC.
- The 2nd order finite difference scheme is exactly same as Crank-Nicolson.

Parabolic Partial Differential Equation

Method of Lines (continued)

- The 4th order finite difference scheme uses 5-point center difference at interior nodes, $x_j, j = 2, \cdots, J-2$.
  - At near-boundary nodes, x_1 and x_{J-1}, alternative high order finite difference formula is used.
  - Boundary nodes x_0 and x_J are governed by boundary conditions.
  - Only interior nodes $x_1, \cdots, x_{J-1}$ are solved from the ODE by implicit IDeC.
  - At each step, we solve a banded diagonal linear system of bandwidth 5.
- When required, we use this 4th order accurate method of lines to smooth out initial values for all other PDE solvers.

Parabolic Partial Differential Equation

Events and Resets

- Apart from the initial smoothing time, Mathwrist PDE solvers also support events and resets.
- Users can specify a list of events. We ensure PDE marching steps will stop at those events.
- Users can instruct a solver that $v(t, x)$ needs be reset at each time marching step. We ensure $v(t, x)$ is updated by calling user supplied callback function.
- Our API function is designed not only to take initial values, but also serve as callbacks at events and resets.
- Updated $v(t, x)$ then becomes the new initial value to continue the process.

References I

[1] Richard L. Burden, J. Douglas Faires and Annette M. Burden: Numerical Analysis, 10th edition, Cengage Learning, 2016

[2] Pedro E. Zadunaisky: On the Estimation of Errors Propagated in the Numerical Integration of Ordinary Differential Equations, Numer. Math. 27, 21 - 39 (1976), Springer-Verlag 1976

[3] Alok Dutt, Leslie Greengard, Vladimir Rokhlin: Spectral Deferred Correction Methods for Ordinary Differential Equations, BIT, 40(2):241–266, 2000

[4] Anders C. Hansen, John Strain: On the Order of Deferred Correction, Applied Numerical Mathematics, Volume 61, Issue 8, August 2011, Pages 961-973

[5] C. Canuto, M.Y. Hussaini, A. Quarteroni and T. A. Zang: Spectral Methods, Fundamentals in Single Domains, Springer, 2006

References II

[6] John P. Boyd: Chebyshev and Fourier Spectral Methods, Dover, 2nd edition, 2016

[7] J. W. Thomas: Numerical Partial Differential Equations: Finite Difference Methods, Springer, 2010

[8] Sandro Salsa: Partial Differential Equations in Action, from Modelling to Theory, Springer 2008