Numerical Relativity: the numerics

Thanasis Giannakopoulos

January 20, 2023

University of Nottingham
UK | CHINA | MALAYSIA

Today

1. Physical problem
2. Formulation
3. PDE analysis
4. **Numerical methods** (method of lines, finite differences)
5. **Implementation** (hyperbolic toy models in `Julia`)
6. **Evaluate errors** (convergence tests)
7. Physical interpretation

References:
- Gustafsson, Kreiss, Oliger: Time Dependent Problems and Difference Methods
- Kreiss, Lorenz: Initial-Boundary Value Problems and the Navier-Stokes Equations
- Sarbach, Tiglio: Continuum and Discrete Initial-Boundary Value Problems and Einstein's Field Equations

From continuum to discrete

Goal: To obtain a numerical approximation that converges to the continuum solution with increasing resolution.

- Choose a numerical scheme that is consistent and stable:
  - consistency: the numerical scheme approximates the correct PDE problem at the continuum limit
  - stability: the solution is controlled by the given data
- Convergence: the difference between the numerical and the continuum solution tends to zero with increasing resolution
  - the convergence rate is related to the numerical scheme

In applications, we typically check numerical convergence.

Semi-discretization: the method of lines

Example PDE: $\partial_t u(t,x) = \partial_x u(t,x)\,, \quad t \geq 0\,, \quad x \in [0,1]$

- spatial (here uniform) grid with spacing $h = \frac{1}{N}$, such that $x = (x_0, x_1, x_2, \ldots, x_N) = (0, h, 2h, \ldots, Nh)$
- Continuum function $u(t,x) \to u_i(t)$ grid function, at each x_i
- $\partial_x \to D_x$, with e.g. $D_x u_i = \frac{u_{i+1} - u_{i-1}}{2h}$

The original example PDE $\to$ a set of coupled ODEs:

$$\partial_t u_i(t) = [u_{i+1}(t) - u_{i-1}(t)]\,\frac{1}{2h}\,, \quad t \geq 0\,, \quad x_0 < x_i < x_N$$

- special care for x_0 and x_N (e.g. different choice of D_x)
- we can use ODE integrators along the line of each x_i
  - a commonly used ODE integrator: 4th order Runge-Kutta (RK4)

Finite difference (FD) operators

Finite difference operators can be derived via Taylor expansions[1].

2nd order accurate:

- forward: $D_x u_i = \frac{-u_{i+2}+4u_{i+1}-3u_i}{2h} + O(h^2)$
- backward: $D_x u_i = \frac{3u_i-4u_{i-1}+u_{i-2}}{2h} + O(h^2)$

- truncation error $O(h^2)$ matched:

- centered: $D_x u_i = \frac{u_{i+1}-u_{i-1}}{2h} + O(h^2)$
- forward: $D_x u_i = \frac{u_{i+3}-4u_{i+2}+7u_{i+1}-4u_i}{2h} + O(h^2)$
- backward: $D_x u_i = \frac{4u_i-7u_{i-1}+4u_{i-2}-u_{i-3}}{2h} + O(h^2)$

Forward and backward FDs can be used to treat the boundaries.

[1] see e.g. Chapter 1 of Pretorius' PhD Thesis

More on numerical methods

Other discretization options:

- Spectral methods

Numerical boundary conditions:

- Ghost points

Numerical stability:

- Use artificial dissipation to damp high frequency noise
- Tune the timestep Δt such that the ODE integrator is stable
  - see Courant-Friedrichs-Lewy (CFL) condition

Numerical convergence (error evaluation)

Assume a convergent numerical scheme (e.g. finite differences) of accuracy n. Then $f - f_h = O(h^n)$ is the numerical error, where:

- continuum solution f
- numerical approximation f_h at resolution h

Take the coarse, medium and fine grid spacings h_c, h_m, h_f. Construct the convergence factor:

$$Q \equiv \frac{h_c^n - h_m^n}{h_m^n - h_f^n} = \frac{f_c - f_m}{f_m - f_f}$$

Example:

- $h_m = h_c/2$, $h_f = h_c/4$ and $n = 2 \rightarrow Q = 4$
- monitor $f_c - f_m$, $f_m - f_f$ and compare with the expected Q

Example: toy models

The PDE problem:

$$\partial_t \phi = \partial_x \phi + \boxed{\partial_x \psi}, \quad \partial_t \psi = \partial_x \psi \,,$$
$$t \in [0,3]\,, \quad x \in [0,1] \text{ with periodic boundary conditions}\,,$$
$$\text{Initial data: } \phi(0,x) = \hat{\phi}(x)\,, \quad \psi(0,x) = \hat{\psi}(x)$$

- ~~$\partial_x \psi$~~ $\rightarrow$ strongly hyperbolic system $\rightarrow$ well-posed IVP in the L^2-norm
- $\partial_x \psi \rightarrow$ weakly hyperbolic system $\rightarrow$ ill-posed IVP in the L^2-norm

$$||\mathbf{u}||_{L^2} = \int \left(\phi^2 + \psi^2\right) dx\,, \quad \mathbf{u} = (\phi, \psi)^T$$

Implementation:

- $\partial_x f \rightarrow D_x f_i$ with 2nd order accurate FD
- ODE integrator is RK4
- timestep $\Delta t = 0.25h$, where h is the spatial grid spacing