MA3227 Numerical Analysis II

Lecture 19: Time-dependent PDEs

Simon Etter

2019/2020

Time-dependent PDEs

Disclaimer

The material in this lecture is quite advanced and will therefore not be part of the homework assignments or exam. The reason why I present it anyway is because it provides a real-world example of how stiff ODEs arise and why it is important to handle such ODEs using implicit Runge-Kutta methods.

Time-dependent PDEs

Introduction

In Lecture 1, we derived the Poisson equation $-\Delta u = f$ from the more general equation

$$\frac{\partial u}{\partial t} = \Delta u + f$$

by setting the time derivative $\frac{\partial u}{\partial t}$ to zero. Poisson's equation therefore describes the steady-state in a diffusion process.

Now that we know how to solve both PDEs and ODEs, we can now also tackle the above time-dependent problem, but before we do let us write out the precise problem statement.

Heat equation

Given $f, u_0 : [0,1]^d \to \mathbb{R}$, find $u : [0,1]^d \times [0, T] \to \mathbb{R}$ such that for all $x \in [0,1]^d$, $x' \in \partial[0,1]^d$ and $t \in [0, T]$ it holds

$$\begin{cases} \frac{\partial u}{\partial t}(x,t) = \Delta u + f, & \text{(PDE)} \\ u(x,0) = u_0(x), & \text{(initial conditions)} \\ u(x',t) = 0. & \text{(boundary conditions)} \end{cases}$$

I will assume $f(x) = 0$ in the following for simplicity.

Time-dependent PDEs

Discretisation of time-dependent PDEs

Most numerical methods for solving time-dependent PDEs are derived according to the following outline.

0. Original PDE:

$$\text{Find } u : [0,1]^d \times [0,T] \to \mathbb{R} \text{ such that } \tfrac{\partial u}{\partial t}(x,t) = \Delta u(x,t). \tag{1}$$

1. Discretise in space using finite differences. This replaces (1) with

$$\text{Find } u : [0,T] \to \mathbb{R}^N \text{ such that } \tfrac{\partial u}{\partial t}(t) = \Delta_n^{(d)}\, u(t). \tag{2}$$

   where $u(t) \in \mathbb{R}^N$ is the vector of point values for finite differences.

2. Discretise in time using a Runge-Kutta method.
   This replaces (2) with

$$\text{Find } (u_\ell \in \mathbb{R}^N)_{\ell=0}^m \text{ such that } u_{\ell+1} = u_\ell + B\, u_\ell\, \tfrac{T}{m}.$$

   where $u_\ell \approx u(T\,\frac{\ell}{m})$.

In this lecture, m denotes the number of temporal grid points, n denotes the number of grid points in each of the d spatial directions, and $N = n^d$ denotes the total number of spatial grid points.

Time-dependent PDEs

Example: finite difference and Euler discretisation of heat equation

0. Original PDE: $\frac{\partial u}{\partial t}(x,t) = \Delta u(x,t)$
1. Finite difference discretisation in space: $\frac{\partial u}{\partial t}(t) = \Delta_n^{(d)} u(t)$ where $u : [0, T] \to \mathbb{R}^N$ is now a time-dependent vector of point values.
2. Explicit Euler time-stepping in time:

$$u\big(\tfrac{\ell+1}{m}\big) \approx u\big(\tfrac{\ell}{m}\big) + \Delta_n^{(d)}\, u\big(\tfrac{\ell}{m}\big)\, \tfrac{T}{m}.$$

See `example()` in `19_time_dependent_pdes.jl`.

Outlook

This lecture follows the usual pattern: we have now a numerical method which appears to be working, so next we will analyse the convergence properties of the proposed method.

Time-dependent PDEs

Convergence analysis for time-dependent PDE solvers

We will use the same notation as in Lecture 16:

- $\Phi : \big(u(x,0), t\big) \mapsto u(x,t)$ denotes the exact one-step propagator.
- $\tilde\Phi : \big(u(x_k,0), t\big)_{k=1}^n \mapsto \big(\tilde u(x_k,t)\big)_{k=1}^n$ denotes the numerical one-step propagator.

Note that Φ maps functions to functions, while $\tilde\Phi$ maps point-values to point-values. As before, we assume that functions are implicitly converted to point-values if this is required by context.

The main new ingredient in time-dependent ODEs is that the numerical propagator involves discretisation errors in both space and time. This requires us to adapt the definition of consistency as follows.

Def: consistency of numerical time propagator

We say that $\tilde\Phi(u)$ is pth-order consistent in space and $(q+1)$th-order consistent in time if

$$\big\|\tilde\Phi(u,t) - \Phi(u,t)\big\|_{2,n} = \mathcal{O}\big(t\,(n^{-p} + t^q)\big).$$

$\|u\|_{2,n} = \sqrt{\frac{1}{(n+1)^d}\sum_{k_1,\ldots,k_d=1}^{n} u[k_1,\ldots,k_d]^2}$ denotes the weighted 2-norm from Lecture 2.

Time-dependent PDEs

Convergence theorem for abstract time-dependent PDE solvers

Assume $\tilde{u}(T)$ is computed using an equidistant temporal mesh $t_\ell = T\,\frac{\ell}{m}$ and a numerical one-step propagator $\tilde{\Phi}(u,t)$ which is pth-order consistent in space and $(q+1)$th-order consistent in time. Then,

$$\|\tilde{u}(T) - u(T)\|_{2,n} = \mathcal{O}\big(n^{-p} + m^{-q}\big).$$

Proof. Combining the error bound from Lecture 16 with the above consistency estimate, we obtain

$$\begin{aligned}
\big\|\tilde{u}_m - u_m\big\|_{2,n} &\leq \sum_{\ell=1}^{m} \exp\big(L\,(t_m - t_\ell)\big)\,\|\tilde{\Phi}_\ell(\tilde{u}_{\ell-1}) - \Phi_\ell(\tilde{u}_{\ell-1})\|_{2,n} \\
&\leq m\,\exp\big(L\,(t_m - t_0)\big)\,\mathcal{O}\Big(\tfrac{T}{m}\,\big(n^{-p} + (\tfrac{T}{m})^q\big)\Big) \\
&= \exp\big(LT\big)\,\mathcal{O}\Big(T\,\big(n^{-p} + (\tfrac{T}{m})^q\big)\Big).
\end{aligned}$$

This proof involves a white lie in that applying the exact time-propagator to the discrete solution, i.e. $\Phi_\ell(\tilde{u}_{\ell-1})$, is not well-defined. Overcoming this issue is non-trivial, but it turns out that we will reach the correct conclusions even with this incorrect argument, so the above is the argument that we will pursue.

Time-dependent PDEs

Consistency of FD + explicit Euler

$$\tilde{u}(t) = u(0) + \Delta_n^{(d)} u(0)\, t$$
$$u(t) = u(0) + \tfrac{\partial u}{\partial t}(0)\, t + \mathcal{O}(t^2)$$

Since

$$\tfrac{\partial u}{\partial t}(x_k, 0) = \Delta u(x_k, 0) = \big(\Delta_n^{(d)} u(0)\big)[k] + \mathcal{O}\big(n^{-2}\big),$$

we have $\tilde{u}(t) - u(t) = \mathcal{O}\big(t\,(n^{-2} + t)\big)$.

This is again a very rough argument where I skip over a lot of details since they will not affect the final result.

Conclusion

Error for finite difference and explicit Euler discretisation with m equidistant time steps and n grid points in each spatial direction is given by

$$\|\tilde{u}(T) - u(T)\|_{2,n} = \mathcal{O}\big(n^{-2} + m^{-1}\big).$$

Note that the spatial order of convergence $p = 2$ is exactly that of the time-independent finite difference method, and the temporal order of convergence $q = 1$ is exactly that of the explicit Euler method.

Time-dependent PDEs

Discussion

Solving a time-dependent PDE involves two limits $m, n \to \infty$. The above analysis shows that the error introduced by truncating both limits can effectively be decomposed into the sum of the errors introduced by truncating either limit.

The practical consequence of this analysis is as follows. If the temporal error component $\mathcal{O}(m^{-q})$ is much smaller than the spatial error component $\mathcal{O}(n^{-p})$, then we could save compute time without losing much in accuracy by reducing m. Conversely if $\mathcal{O}(m^{-q})$ is much larger than $\mathcal{O}(n^{-p})$, then we could save compute time without losing much in accuracy by reducing n.

Optimal efficiency is thus achieved if the two error components are of roughly equal magnitude, i.e. if $m^{-p} \approx n^{-q}$.

Concrete examples (assuming finite difference discretisation in space):

- For explicit or implicit Euler, choose $m \propto n^2$.
- For explicit or implicit midpoint, choose $m \propto n$.

See `convergence()` for numerical demonstration.

Time-dependent PDEs

Discussion (continued)

Another phenomenon which can be observed in `convergence()` is that the error blows up if we set `step` to an explicit method and m is much smaller than n.

This observation is due to the stability phenomenon described in Lecture 18 as we shall see next.

Fixed point of the heat equation

Recall that the heat equation $\frac{\partial u}{\partial t} = \Delta u$ with boundary conditions $u(\partial\Omega, t) = 0$ describes diffusion in a domain $\Omega \subset \mathbb{R}^d$ where the temperature / concentration of particles at the boundary is always kept at 0. Physical intuition therefore tells us that if we wait long enough, then the temperature / concentration $u(x, t)$ will go to 0 everywhere on Ω, i.e. $u_F(x) = 0$ is an attractive fixed point of the system.

Time-dependent PDEs

Fixed point of the heat equation (continued)

The physical intuition can be easily confirmed for the discretised-in-space ODE $\frac{\partial u}{\partial t} = \Delta_n^{(d)} u$: assume $V \Lambda V^T = \Delta_n^{(d)}$ is the eigendecomposition of $\Delta_n^{(d)}$, and set $w = V^T u$. Then,

$$\dot{w} = V^T \dot{u} = V^T \Delta_n^{(d)} u = V^T V \Lambda V^T u = \Lambda V^T u = \Lambda w$$

and hence

$$w(t) = \exp(\Lambda t)\, w_0, \qquad u(t) = V w(t) = V \exp(\Lambda t)\, w_0 = V \exp(\Lambda t) V^T u_0$$

where $\exp(\Lambda t)$ denotes the diagonal matrix defined by

$$\exp(\Lambda t)[i,i] = \exp(\Lambda[i,i]\, t).$$

We know that all eigenvalues $\Lambda[i,i]$ of $\Delta_n^{(d)}$ are negative; hence $\exp(\Lambda t) \to 0$ for $t \to 0$ as expected.

Time-dependent PDEs

Discussion

We are now in precisely the situation discussed in Lecture 18: we know that the exact solution $u(k\,\Delta t)$ vanishes exponentially for $k \to \infty$, but the numerical solution $\tilde{u}(k\,\Delta t)$ will vanish only if $|R(\lambda\Delta t)| < 1$ for all eigenvalues λ of $\Delta_n^{(d)}$.

We have further seen:

- For explicit Euler and midpoint and $\lambda < 0$, we have

$$|R(\lambda\Delta t)| < 1 \quad \iff \quad \Delta t < \tfrac{2}{-\lambda}.$$

- The eigenvalue of largest absolute value $\lambda_{\max}$ of $\Delta_n^{(d)}$ satisfies $\lambda_{\max} \approx -4d\,(n+1)^2$.

Thus, since $\Delta t < \frac{2}{-\lambda}$ must hold for all eigenvalues λ, we conclude that we must satisfy the step size constraint

$$\Delta t < \big(2d\,(n+1)^2\big)^{-1} \quad \implies \quad m \geq C\,n^2$$

to avoid spurious exponential blow-up of the numerical solution.

This explains our numerical observations in `convergence()`.

Time-dependent PDEs

Discussion (continued)

The time step constraint $m > C\,n^2$ is acceptable for the explicit Euler method since convergence anyway requires $m \propto n^2$.

On the other hand, the time step constraint $m > C\,n^2$ implies that the explicit midpoint method offers no advantage over the explicit Euler method: for accuracy purposes, $m \propto n$ should be enough, but stability imposes $m \propto n^2$.

Conclusion: when solving a time-dependent PDE, we should either use the explicit Euler method or an implicit Runge-Kutta scheme.

Time-dependent PDEs

Summary

- Time-dependent PDEs can be solved by first discretising in space and then applying a Runge-Kutta scheme to the resulting system of ODEs.
- Stability constraints imply that only the explicit Euler method or an implicit Runge-Kutta scheme can solve the system of ODEs effectively.