The 2D Heat Equation
— Analysis and Numerical Approaches

Université de Paris

Outline

An Evolutional Problem

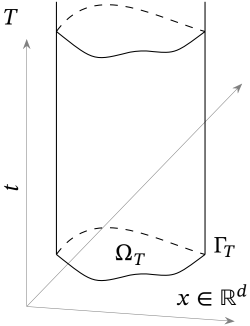


Figure: Region Ω_T

- Let $\Omega \subset \mathbb{R}^d$ be an open set with boundry $\Gamma := \partial\Omega$
- Set $\Omega_T = \Omega \times]0, T[$, $\Gamma_T := \Gamma \times]0, T[$, Γ_T is called the *lateral* boundary of the cylinder Ω_T.
- Consider the heat equation with τ-periodic boundary condition:

$$\begin{cases} \partial_t u - \Delta_x u = f & \text{on } \Omega_T \\ u(x_i, t) = u(x_i + \tau, t) & \text{on } \Gamma_T \\ u(x, 0) = u_0(x) & \text{on } \Omega \times \{t = 0\} \end{cases}$$

Outline

Schemes based on FDM

[cf. Luc16, ch.08]

- The Explicit Euler Three Point Finite Difference Scheme
- The Implicit Euler and Leapfrog Schemes
- The Crank-Nicolson Scheme

Schemes based on FDM

[cf. Luc16, ch.08]

- The Explicit Euler Three Point Finite Difference Scheme
- The Implicit Euler and Leapfrog Schemes
- The Crank-Nicolson Scheme

Schemes based on FDM

[cf. Luc16, ch.08]

- The Explicit Euler Three Point Finite Difference Scheme
- The Implicit Euler and Leapfrog Schemes
- The Crank-Nicolson Scheme

Meshes in 2D

[cf. Luc16, ch.06; Zil21]

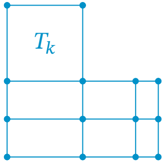
- Rectangular Q_1 Finite Elements
- Triangular P_1 and P_2 Lagrange Elements

(a) Rectangular Q_1 Finite Elements (b) Triangular P_1 Lagrange Elements (c) Triangular P_2 Lagrange Elements

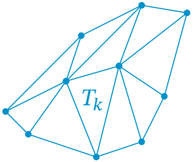
Meshes in 2D

[cf. Luc16, ch.06; Zil21]

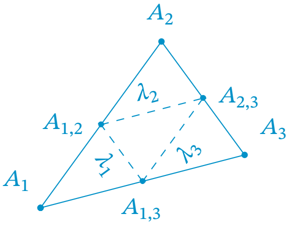
- Rectangular Q_1 Finite Elements
- Triangular P_1 and P_2 Lagrange Elements



(a) Rectangular Q_1 Finite Elements



(b) Triangular P_1 Lagrange Elements



(c) Triangular P_2 Lagrange Elements

The Riesz-Fredholm Theory

[cf. Bre11, ch.06; Sch01, ch.08]

Outline

[cf. Luc16; Zil21]

All algorithms are implemented by Python 3.9 on 64-bit machine.

- Assemble with sparse matrix: `import scipy.sparse`
- JIT (Just-In-Time) compile: `import numba`
- Parallel computation for large volume data processing

- Assemble with sparse matrix: `import scipy.sparse`
- JIT (Just-In-Time) compile: `import numba`
- Parallel computation for large volume data processing

- Assemble with sparse matrix: `import scipy.sparse`
- JIT (Just-In-Time) compile: `import numba`
- Parallel computation for large volume data processing

Outline

Université de Paris

The Black-Scholes PDE for the Pricing of Financial Derivatives

[cf. Nef00, ch.12-13]

$$
\begin{aligned}
-rF + \partial_t F + \partial_t F \partial_t S + \frac{\sigma_t^2}{2} \Delta_s F \partial_t^2 S = 0 \\
S_t \in [0, +\infty[, \quad t \in [0, T] \\
F(T) = (S_T - K)_+
\end{aligned}
$$

References:

[Bre11] Haim Brezis. *Functional Analysis, Sobolev Spaces and Partial Differential Equations*. Springer New York, 2011.

[Eva10] Lawrence Evans. *Partial Differential Equations*. American Mathematical Society, Mar. 2010.

[Luc16] Hervé Le Dret ; Brigitte Lucquin. *Partial Differential Equations: Modeling, Analysis and Numerical Approximation*. Springer International Publishing, 2016.

[Nef00] Salih N. Neftci. *An Introduction to the Mathematics of Financial Derivatives*. Elsevier Science & Techn., June 2000. 527 pp.

[Sch01] Michelle Schatzman. *Analyse numérique : une approche mathématique : cours et exercices*. Paris: Dunod, 2001.

[Zil21] Matthieu Bonnivard; Adina Ciomaga; Alessandro Zilio. "Méthodes numériques pour les EDO et les EDP". Notes de cours M1 Mathématiques. 2021.

Université de Paris

Thank you for listening!

Questions?

All resources including this diapositive, the final report, code and user's manual are submitted to:

Project Github: https://github.com/derrring/2d_heat_equation

Welcome to give us your valuable feedbacks :)

Outline

Université de Paris