Computational Finance FIM 201 (3 ECTS)

Instructor: Dr. Selim Gökay (selimgokay2003@gmail.com)

Basics of the Course:

Content: To explore numerical methods for options.

Focus:

- Monte Carlo Simulations.
- Finite Difference Methods (FDM).

Approach:

- Rigorous mathematics class
- Together with hands-on applications.

Programming Language: (Only) Python.

References:

- Tools for Computational Finance, Rudiger U. Seydel
- Monte Carlo Methods in Financial Engineering, Paul Glassermann
- Computational Methods for Option Pricing, Yves Achdou, Olivier Pironneau

Aims of the Course:

- To apply simulations or FDM to compute option prices and sensitivities.
- To pick an appropriate numerical method for the problem at hand.
- To prove the convergence of a chosen numerical method.
- To gain exposure to implement option pricing algorithms in Python.

Prior Knowledge:

- Probability I, Probability II and Mathematical Finance II.
- No prior experience in Python assumed.

Tentative Outline of FIM 201

1

Black-Scholes

European Vanillas

Simulation of Random Variables and Selected Variance Reduction Techniques

2

Models beyond Black-Scholes

Path-dependent Options

“Exact” Simulation of Stochastic Processes and Selected Variance Reduction Techniques

3

Discretization Methods for Stochastic Differential Equations, Euler and Milstein Schemes

4

Black-Scholes

Vanillas, Barriers

Finite Difference Methods Theta-scheme.

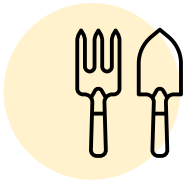
5

Models Beyond Black-Scholes

American Options

American Options via Longstaff-Schwartz method

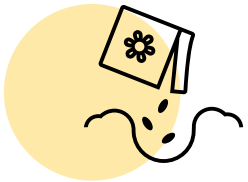
Class Structure of FIM 201



Motivating Problem

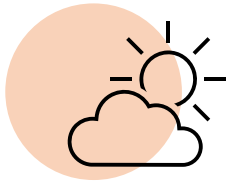
Every class starts with a financial problem.

For example, the first class will illustrate European option pricing in the Black-Scholes model.



Mathematical Analysis

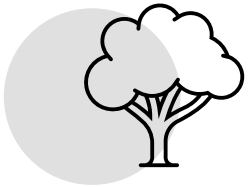
This class will contain rigorous analysis of the algorithms containing all the relevant convergence proofs.



The Algorithm

An appropriate algorithm will be introduced to solve the problem.

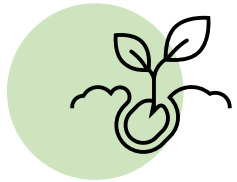
For example, in the first class we will learn how to simulate random variables such as normal distribution to value European vanillas in Black-Scholes model.



Analysis of the algorithms

Advantages and shortcomings of the algorithms in terms of convergence as well as implementation will be discussed.

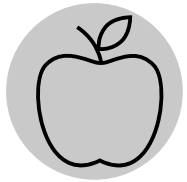
This will help students to choose an appropriate numerical method for the problem at hand.



Implementation via Python

All implementation as well as the HW's will be done only in Python.

No experience in Python is required. The code for all the examples in class will be distributed.



Evaluation via Homework and Final Exam

There will be five or six homework assignments, which are crucial for the class. Students need to be proactive in completing them. Each HW assignment will introduce a similar but slightly different problem as in the lecture. Sample starting codes will be distributed to students that they can play around with to complete the exercises. In addition, there will be a final written exam.