Applications Computational Finance, II

Goals

- Overview of methods to price options
- Directions on how to proceed (resources, research, programming)
- Using C++ code and libraries to ‘bootstrap’ your efforts
- Learning to map algorithms to code

2

Main Methods

- PDE methods (FDM, FEM, meshless, spectral)
- Integral methods (Conv, Cos, Fourier)
- Simulation methods (Monte Carlo, Lattice (binomial, trinomial))
- Exact/closed solutions

Some Use Cases

- Price options (equity, fixed income, interest rate, hybrid)
- Hedging and option sensitivities
- Calibration and parameter estimation

PDE Methods (FDM) (1/2)

- Option model described as a partial differential equation (PDE) defined in some region of (S,t) space
- Can be one-factor or multi-factor equations
- Describe any *contingent claim* that ‘depends’ on some *underlying variable*
- Most PDEs are of *convection-diffusion* type

5

PDE Methods (FDM) (2/2)

- Approximate PDE by finite difference methods (FDM)
- Solve discrete assembled equations by matrix solvers
- We discretise in space and time (different methods)
- Stability and accuracy of finite difference schemes

6

Main FDM Categories

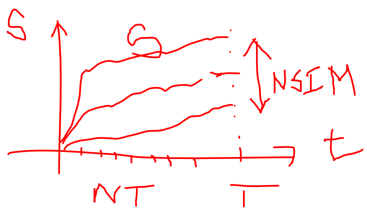
- Alternating Direction Implicit (ADI)
- Method of Fractional Steps ("Soviet Splitting")
- Alternating Direction Explicit (ADE)
- Others, sub-themes and variations

Stochastic Differential Equations (SDE)

- Describe random behaviour of underlying variables (one-factor, n-factor)
- No closed solution in general
- Use FDM to approximate the SDE
- Use SDE as basis for Monte Carlo and lattice methods

Monte Carlo

- A simulation method; based in multiple times doing a 'SDE-FDM' trajectory
- Geometric Brownian Motion (GBM); random walk on [0,T]
- Take multiple samples
- Discount summed values from t = T to t = 0



9

Lattice Methods

- Similar to Monte Carlo in spirit and structure
- We trace evolution of the underlying in time (it can go up or down) up to t = T (*forward induction*)
- We then carry out *backward induction* from t = T to t = 0
- The price at t = 0 is the option price

10

How do we evaluate a Method?

- Applicability to a wide range of derivatives types
- Robustness (works for all parameters), accuracy and efficiency
- Ease of use/implementation
- Maintainability (becomes important when software system starts to stabilise)

FDM

- Accuracy and efficiency are advantages
- Difficulty in setting up the schemes and fine-tuning
- FDM for n-fact0r problems difficult for n = 4, 5, ...
- Ideally, applied/numerical mathematics (or similar) background needed

Monte Carlo

- Applicable to a wide range of option types
- Easy to apply (a *blunt instrument*)
- Difficulty with computation of option sensitivies and early exercise
- Slow convergence, not very accurate method
- Mathematical foundations not as strong as with PDE methods

13

Exact Solutions

- No numerical approximations needed (in principle)
- In a form that is easily computed, in principle
- Only applicable to a given range of the parameter set
- It might forever to come up with a solution

14

FDM Building Blocks

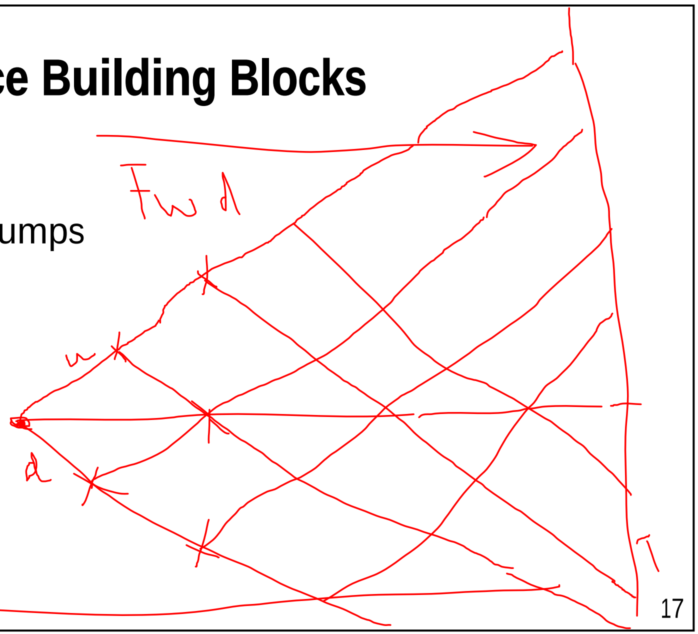
- Creating meshes in S and t
- Vectors and matrices
- Data structures for PDE and FDM entities

MC Building Blocks

- Random number generators
- Data structures for SDE and FDM
- Vectors and matrices
- Payoff functions

Lattice Building Blocks

- Lattice data structures
- Algorithms to compute jumps
- Payoff entities



Bwd

17