Two Factor FDM

Lecture 04

Fall 2025
KAIST MFE
금융수치해석기법

Two factor FDM 예제

Two-factor model이 사용되는 예시

- 전환사채의 가격 요인
  - 기초자산 주가
  - Default or interest rate 요인
- 2 요인에 의해 가격이 결정되는 옵션은 FDM으로 용이하게 평가 가능, but 3 요인 이상에서는 계산량이 급증하여 FDM의 효율성이 급격히 낮아짐 → 다요인 모델은 몬테카를로 시뮬레이션을 이용

예시

- Stochastic volatility model을 이용한 옵션 가격 평가
- 기초자산이 2개인 multi-asset 옵션의 평가
- Two-factor term structure model을 이용한 금리 파생상품 평가

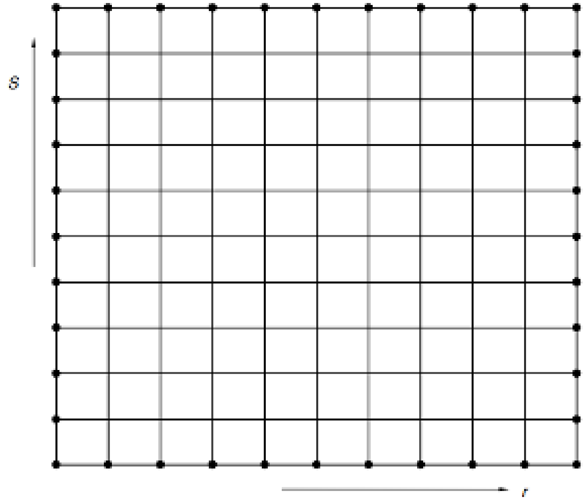
PDE for two-factor equation:

$$V_t + a(S,r,t)V_{SS} + b(S,r,t)V_S + c(S,r,t)V$$
$$+d(S,r,t)V_{rr} + e(S,r,t)V_{Sr} + f(S,r,t)V_r = 0$$

- Grid: 3차원(주가, 금리, 시간) 그리드 필요

$$S = i\delta S, \quad r = j\delta r, \quad t = k\delta t$$

Terminal condition / Cross-derivatives approximation



Contract value at node (*i,j*):

- $V(S,r,t) = V_{ij}^k$

Terminal condition:

- $V_{ij}^N = \max(S_i - E, 0)$
- E: 전환가격

Second derivatives with respect to both S and r

$$\frac{\partial^2 V}{\partial S \partial r} = \frac{\partial}{\partial S}\left(\frac{\partial V}{\partial r}\right) \approx \frac{\frac{\partial V}{\partial r}(S+\delta S, r, t) - \frac{\partial V}{\partial r}(S-\delta S, r, t)}{2\delta S}$$

$$= \frac{\frac{V_{i+1,j+1}^k - V_{i+1,j-1}^k}{2\delta r} - \frac{V_{i-1,j+1}^k - V_{i-1,j-1}^k}{2\delta r}}{2\delta S} = \frac{V_{i+1,j+1}^k - V_{i+1,j-1}^k - V_{i-1,j+1}^k + V_{i-1,j-1}^k}{4\delta S \delta r}$$

Explicit difference scheme

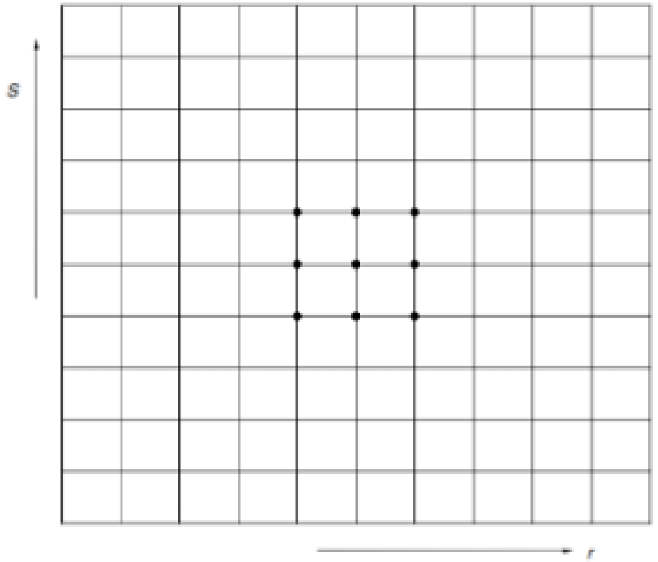
$$V_t + a(S,r,t)V_{SS} + b(S,r,t)V_S + c(S,r,t)V + d(S,r,t)V_{rr} + e(S,r,t)V_{Sr} + f(S,r,t)V_r = 0$$

$$\Rightarrow \frac{V_{ij}^k - V_{ij}^{k-1}}{\delta t} + a_{ij}^k \left(\frac{V_{i+1,j}^k - 2V_{ij}^k + V_{i-1,j}^k}{\delta S^2} \right) + b_{ij}^k \left(\frac{V_{i+1,j}^k - V_{i-1,j}^k}{2\delta S} \right) + c_{ij}^k V_{ij}^k$$

$$+ d_{ij}^k \left(\frac{V_{i,j+1}^k - 2V_{ij}^k + V_{i,j-1}^k}{\delta r^2} \right) + e_{ij}^k \left(\frac{V_{i+1,j+1}^k - V_{i+1,j-1}^k - V_{i-1,j+1}^k + V_{i-1,j-1}^k}{4\delta S \delta r} \right)$$

$$+ f_{ij}^k \left(\frac{V_{i,j+1}^k - V_{i,j-1}^k}{2\delta r} \right) = O(\delta t, \delta S^2, \delta r^2).$$

- 9 points used to time step in the explicit method.



Two-dimensional Black-Scholes Equation

Black-Scholes Equation:

$$\mathcal{L}_{BS} = \frac{1}{2}\sigma_1^2 x^2 v_{xx} + \frac{1}{2}\sigma_2^2 y^2 v_{yy} + \rho\sigma_1\sigma_2 xy v_{xy} + rxv_x + ryv_y - rv$$

$$-v_t = \mathcal{L}_{BS}$$

FDM Grid: $\Omega \times [0, T]$

$$\Omega = \{(x, y) : x > 0, y > 0\}$$

- N_x, N_y, N_t steps
- Step size: $h_x = L/N_x, \quad h_y = M/N_y$
- Time step $\Delta t = T/N_t$
- $u_{ij}^n = u(ih, jh, n\Delta t)$

Boundary condition

- Linear boundary condition

$$u_{xx}(0, y, t) = u_{xx}(L, y, t)$$
$$= u_{yy}(x, 0, t) = u_{yy}(x, M, t) = 0$$

Explicit FDM

$$\frac{v_{ij}^{n-1} - v_{ij}^{n}}{\Delta t}$$

$$= \frac{1}{2}\sigma_x^2 x_i^2 \frac{v_{i+1,j}^n - 2v_{ij}^n + v_{i-1,j}^n}{h_x^2} + rx_i \frac{v_{i+1,j}^n - v_{i-1,j}^n}{2h_x} + \frac{1}{2}\sigma_y^2 y_j^2 \frac{v_{i,j+1}^n - 2v_{ij}^n + v_{i,j-1}^n}{h_y^2} + ry_j \frac{v_{i,j+1}^n - v_{i,j-1}^n}{2h_y}$$

$$+ \rho\sigma_x\sigma_y x_i y_j \frac{v_{i+1,j+1}^n - v_{i+1,j-1}^n - v_{i-1,j+1}^n + v_{i-1,j-1}^n}{4h_x h_y} - rv_{ij}^n$$

$$\Downarrow$$

$$v_{ij}^{n-1}$$

$$= \Delta t \left(\frac{\sigma_x^2 x_i^2}{2h_x^2} + \frac{rx_i}{2h_x} \right) v_{i+1,j}^n + \Delta t \left(\frac{\sigma_x^2 x_i^2}{2h_x^2} - \frac{rx_i}{2h_x} \right) v_{i-1,j}^n + \Delta t \left(\frac{\sigma_y^2 y_j^2}{2h_y^2} + \frac{ry_j}{2h_y} \right) v_{i,j+1}^n + \Delta t \left(\frac{\sigma_y^2 y_j^2}{2h_y^2} - \frac{ry_j}{2h_y} \right) v_{i,j-1}^n$$

$$+ \frac{\Delta t \rho\sigma_x\sigma_y x_i y_j}{4h_x h_y} (v_{i+1,j+1}^n - v_{i+1,j-1}^n - v_{i-1,j+1}^n + v_{i-1,j-1}^n) + \left(1 - \Delta t \left(\frac{\sigma_x^2 x_i^2}{h_x^2} + \frac{\sigma_y^2 y_j^2}{h_y^2} + r \right)\right) v_{ij}^n$$

Alternating Direction Implicit (ADI)

- 하나의 time step을 2단계로 구분하고, 1단계에서는 x에 대해 implicit, y에 대해서는 explicit방법을 적용하고, 2단계에서는 반대로 x에 대해 explicit, y에 대해 implicit 방법을 순차적으로 적용함

$$\frac{v_{ij}^{n-0.5} - v_{ij}^{n}}{\Delta t/2} = \mathcal{L}_{ADI}^{x} v_{ij}^{n-0.5} + \mathcal{L}_{ADI}^{y} v_{ij}^{n}$$

$$\frac{v_{ij}^{n-1} - v_{ij}^{n-0.5}}{\Delta t/2} = \mathcal{L}_{ADI}^{x} v_{ij}^{n-0.5} + \mathcal{L}_{ADI}^{y} v_{ij}^{n-1}$$

ADI Discretization

- 1 단계

$$
\begin{aligned}
&\mathcal{L}^x_{ADI} v^{n-0.5}_{ij} \\
&= \frac{1}{2}\sigma_x^2 x_i^2 \frac{v^{n-0.5}_{i+1,j} - 2v^{n-0.5}_{ij} + v^{n-0.5}_{i-1,j}}{h_x^2} + r x_i \frac{v^{n-0.5}_{i+1,j} - v^{n-0.5}_{i-1,j}}{2h_x} + \frac{1}{2}\sigma_y^2 y_j^2 \frac{v^{n}_{i,j+1} - 2v^{n}_{ij} + v^{n}_{i,j-1}}{h_y^2} + r y_j \frac{v^{n}_{i,j+1} - v^{n}_{i,j-1}}{2h_y} \\
&+ \rho\sigma_x\sigma_y x_i y_j \frac{v^{n}_{i+1,j+1} - v^{n}_{i+1,j-1} - v^{n}_{i-1,j+1} + v^{n}_{i-1,j-1}}{4h_x h_y} - r v^{n-0.5}_{ij}
\end{aligned}
$$

- 2 단계

$$
\begin{aligned}
&\mathcal{L}^y_{ADI} v^{n-1}_{ij} \\
&= \frac{1}{2}\sigma_x^2 x_i^2 \frac{v^{n-0.5}_{i+1,j} - 2v^{n-0.5}_{ij} + v^{n-0.5}_{i-1,j}}{h_x^2} + r x_i \frac{v^{n-0.5}_{i+1,j} - v^{n-0.5}_{i-1,j}}{2h_x} + \frac{1}{2}\sigma_y^2 y_j^2 \frac{v^{n-1}_{i,j+1} - 2v^{n-1}_{ij} + v^{n-1}_{i,j-1}}{h_y^2} + r y_j \frac{v^{n-1}_{i,j+1} - v^{n-1}_{i,j-1}}{2h_y} \\
&+ \rho\sigma_x\sigma_y x_i y_j \frac{v^{n-0.5}_{i+1,j+1} - v^{n-0.5}_{i+1,j-1} - v^{n-0.5}_{i-1,j+1} + v^{n-0.5}_{i-1,j-1}}{4h_x h_y} - r v^{n-1}_{ij}
\end{aligned}
$$

Operator Splitting Method (OSM)

- 여러 개의 단순한 연산으로 나눠서 해결
- x에 대한 연산을 우선 적용하여 $v_{ij}^{n-0.5}$를 계산하고, 이후에 y에 대한 연산을 적용하여 v_{ij}^{n-1}을 구함
- 교차항은 explicit 방법으로 처리

$$\frac{v_{ij}^{n-0.5} - v_{ij}^{n}}{\Delta t} = \mathcal{L}_{OS}^{x} v_{ij}^{n-0.5}$$

$$\frac{v_{ij}^{n-1} - v_{ij}^{n-0.5}}{\Delta t} = \mathcal{L}_{OS}^{y} v_{ij}^{n-1}$$

OSM Discretization

- x 연산

$$\mathcal{L}_{OS}^{x} v_{ij}^{n-0.5} = \frac{1}{2}\sigma_x^2 x_i^2 \frac{v_{i+1,j}^{n-0.5} - 2v_{ij}^{n-0.5} + v_{i-1,j}^{n-0.5}}{h_x^2} + r x_i \frac{v_{i+1,j}^{n-0.5} - v_{i-1,j}^{n-0.5}}{2h_x} - \frac{1}{2} r v_{ij}^{n-0.5} + \frac{1}{2}\rho\sigma_x\sigma_y x_i y_j \frac{v_{i+1,j+1}^{n} - v_{i+1,j-1}^{n} - v_{i-1,j+1}^{n} + v_{i-1,j-1}^{n}}{4h_x h_y}$$

- y 연산

$$\mathcal{L}_{OS}^{y} v_{ij}^{n-1} = \frac{1}{2}\sigma_y^2 y_j^2 \frac{v_{i,j+1}^{n-1} - 2v_{ij}^{n-1} + v_{i,j-1}^{n-1}}{h_y^2} + r y_j \frac{v_{i,j+1}^{n-1} - v_{i,j-1}^{n-1}}{2h_y} - \frac{1}{2} r v_{ij}^{n-1} + \frac{1}{2}\rho\sigma_x\sigma_y x_i y_j \frac{v_{i+1,j+1}^{n-0.5} - v_{i+1,j-1}^{n-0.5} - v_{i-1,j+1}^{n-0.5} + v_{i-1,j-1}^{n-0.5}}{4h_x h_y}$$

Binary Option Pricing

Binary option and Call spread

- Binary option의 델타 헤징을 이용한 복제 전략을 실행할 때, 행사가격 부근에서 gamma가 매우 커지는 현상으로 인해 리스크가 매우 높아짐
- 실무적으로는 이런 리스크를 완화하기 위해서 call spread 구조를 이용한 overhedge(업계에서 통용되는 용어이나 공식적인 용어는 아닌 것 같음)를 실행함

Worst-Of ELS 구조

- 여러 개의 주가지수 기초자산 가운데 발행일 대비 성과가 가장 저조한 지수에 의해 payoff 가 결정됨
- 국내에서 발행되는 ELS의 대표적인 구조는 3년 만기, 6개월 마다 worst performer가 조기행사 베리어 이상인 경우 중도상환, 만기에서 worst performer가 최종 상환 베리어 이하인 경우 손실 상환 구조

과제: Worst-of binary option을 평가하는 FDM 코드 작성(ADI와 OSM 구현)

- 2개의 기초자산의 가격을 S_1과 S_2라고 하고, 발행시점의 가격을 각각 100으로 동일하다고 가정
- 만기 시점의 overhedge를 고려한 binary option의 payoff는 다음과 같음 (oh는 overhedge 파라미터)

$$\text{payoff} = \begin{cases} 1 & \text{if } \min(S_1, S_2) > K \\ \frac{1}{oh}(\min(S_1, S_2) - (K - oh)) & \text{if } K - oh < \min(S_1, S_2) \le K \\ 0 & \text{if } \min(S_1, S_2) \le K - oh \end{cases}$$