1. Introduction

Mono-underlying products are payoffs written on a single name underlying (equity, interest rate, FX rate...). Unlike multi-underlying products, these products might be valued using efficient partial differential equations.

The aim of this project is to value vanilla, european and path dependent products using PDE.

PDE are possible for multi-dimensional problems. However, their resolution suffers from the course of dimensionality. Indeed, Monte Carlo pricers are preferred for multi-dimensional pricing.

In this project, a 1-dimension model is assumed for the underlying dynamics referred to without loss of generality as stock or equity.

2. Mathematical background

2.1 Log normal dynamics

The equity underlying follows a log-normal dynamics given by

$$dS_t = (r_t - q_t)S_t dt - \sum_i d_i 1_{t=\tau_i} + \sigma(t, S_t)S_t dW_t \tag{2.1}$$

where

- r_t is the risk free interest rate
- q_t is the repo rate inclusive of continuous dividend yield
- d_i are discrete dividened cash and proportional paid at τ_i: $d_i = \alpha_i S_{\tau_i^-} + \beta_i$
- $(t, S) \to \sigma(t, S)$ is the local deterministic local volatility function computed using Dupire formula.

For simplicity, we will assume that there are no discrete dividends (the general case could be handled as an extra bonus).

2.2 The pricing equation

As in the derivation of the Black-Scholes model, the starting point is to build a self financed strategy that is delta hedged.

A self financed strategy could set as follow:

- At time t, a 0-value position is built:
  - the option seller (here the Bank) will receive a premium P_t paid by the option buyer (the Client).
  - This amount P_t is passed to treasury desk. The Bank's volatility desk will receive a rate r_t on this cash amount.
  - The desk will buy a quantity δ_t of the underlying of the option.
  - In order to buy the shares, the vanilla desk asks for funding from its treasury / funding desk for an amout $\delta_t S_t$. The vanilla desk will pass the shares to the funding desk who will manage to enter in repo either with internal or external conterparties.

- The repo funding (more advantageous than a non collateralized funding) will be reimboursed by vanilla desk at a rate $r_t - repo(t)$
- The shares will pay a dividend yield $divYield(t)$

- At time $t + dt$, the vanilla desk:
  - receives the cash premium and interest $+P_t \times (1 + r_t dt)$
  - receives any dividend (continuous) paid over the period: $+\delta_t S_t \times divYield(t) \times dt$
  - has shares that are valued $+\delta_t S_{t+dt}$
  - pay interest on the function of the shares $-\delta_t S_t \times (1 + (r_t - repo(t))dt)$
  - is short on the vanilla position that is valued $-P_{t+dt}$

Auto-financement and risk free assumptions ensure that the above portfolio should be worth 0 at $t + dt$. Hence:

$$\begin{aligned}\pi_{t+dt} &= +P_t \times (1 + r_t dt) + \delta_t S_t \times divYield(t) \times dt + \delta_t S_{t+dt} - \delta_t S_t \times (1 + (r_t - repo(t))dt) - P_{t+dt} \\ &= -(P_{t+dt} - P_t) + \delta_t (S_{t+dt} - S_t) + r_t P_t dt - \delta_t S_t (r_t - repo(t) - divYield(t))dt \\ &= -dP_t + \delta_t dS_t + r_t P_t dt - \delta_t S_t (r_t - q_t)dt\end{aligned}$$

where we have used the fact that $q_t = repo(t) + divYield(t)$.

Using Itô forumla, one has:

$$\begin{aligned}dP_t &= \partial_t P dt + \partial_S P dS_t + \frac{1}{2}\partial^2_{SS} P \times d < S_t | S_t > \\ &= \partial_t P dt + \partial_S P dS_t + \frac{1}{2}\sigma^2(t, S_t) S_t^2 \partial^2_{SS} P dt\end{aligned}$$

Then,

$$d\pi_t = -\partial_t P dt - \frac{1}{2}\sigma^2(t, S_t) S_t^2 \partial^2_{SS} P dt + (\delta_t - \partial_S P) dS_t + r_t P_t dt - \delta_t S_t (r_t - q_t) dt$$

In order to hedge S_t risk / exposure, one chooses $\delta_t = \partial_S P$.

The arbitrage free assumption leads to $d\pi_t = 0$.
Finally, the pricing equation follow:

$$\partial_t P + (r_t - q_t) S \partial_S P + \frac{1}{2}\sigma^2(t, S) S^2 \partial^2_{SS} P - r_t P_t = 0 \qquad (2.2)$$

Usually, one introduces the following change of variable to resolve equation (2.2):

$$x_t = ln(\frac{S}{F(0,t)}) = ln(\frac{S}{S_0}) - \int_0^t (r_s - q_s) ds$$

where $F(0,t) = S_0 e^{\int_0^t (r_s - q_s) ds}$ is the forward price of maturity t valued at time 0.

Let $\tilde{P}(t,x) = e^{-\int_0^t r_s ds} P(t,S)$. Then

$$P(t,S) = \tilde{P}(t,x) e^{\int_0^t r_s ds}$$

Derivatives are computed below:

$$\begin{aligned}\partial_t P(t,S) &= \partial_t(\tilde{P}(t,x_t)) \times e^{\int_0^t r_s ds} + r_t e^{\int_0^t r_s ds} \times \tilde{P}(t,x_t) \\ &= (\partial_t(\tilde{P}(t,x_t)) + r_t \times \tilde{P}(t,x_t)) e^{\int_0^t r_s ds} \\ &= (\partial_t \tilde{P}(t,x_t) + \partial_x \tilde{P}(t,x_t) \times \partial_t x_t + r_t \times \tilde{P}(t,x_t)) e^{\int_0^t r_s ds} \\ &= (\partial_t \tilde{P}(t,x_t) - (r_t - q_t) \partial_x \tilde{P}(t,x_t) + r_t \times \tilde{P}(t,x_t)) e^{\int_0^t r_s ds}\end{aligned}$$

$$\begin{aligned}\partial_S P(t,S) &= e^{\int_0^t r_s ds}\partial_S(\tilde{P}(t,x_t))\\ &= e^{\int_0^t r_s ds}\partial_x\tilde{P}(t,x_t)\times\partial_S x_t\\ &= e^{\int_0^t r_s ds}\partial_x\tilde{P}(t,x_t)\times\frac{1}{S}\end{aligned}$$

$$\begin{aligned}\partial^2_{SS}P(t,S) &= \partial_S(e^{\int_0^t r_s ds}\partial_x\tilde{P}(t,x_t)\times\frac{1}{S})\\ &= e^{\int_0^t r_s ds}\partial_S(\partial_x\tilde{P}(t,x_t)\times\frac{1}{S})\\ &= e^{\int_0^t r_s ds}(\partial_S(\partial_x\tilde{P}(t,x_t))\times\frac{1}{S}+\partial_x\tilde{P}(t,x_t)\times\partial_S\frac{1}{S})\\ &= e^{\int_0^t r_s ds}(\partial_x(\partial_x\tilde{P}(t,x_t))\times\partial_S x_t\times\frac{1}{S}-\frac{1}{S^2}\partial_x\tilde{P}(t,x_t))\\ &= e^{\int_0^t r_s ds}(\partial^2_{xx}\tilde{P}(t,x_t)\times\frac{1}{S}\times\frac{1}{S}-\frac{1}{S^2}\partial_x\tilde{P}(t,x_t))\\ &= e^{\int_0^t r_s ds}\frac{1}{S^2}(\partial^2_{xx}\tilde{P}(t,x_t)-\partial_x\tilde{P}(t,x_t))\end{aligned}$$

Finally:

- $\partial_t P(t,S) = e^{\int_0^t r_s ds}(\partial_t\tilde{P}(t,S)-(r_t-q_t)\partial_x\tilde{P}+r_t\tilde{P})$
- $\partial_S P(t,S) = e^{\int_0^t r_s ds}\frac{1}{S}\partial_x\tilde{P}$
- $\partial^2_{SS}P(t,S) = e^{\int_0^t r_s ds}\frac{1}{S^2}(\partial^2_{xx}\tilde{P}-\partial_x\tilde{P})$

Finally, we get an alternative PDE to resolve:

$$\partial_t\tilde{P}(t,S)-(r_t-q_t)\partial_x\tilde{P}+r_t\tilde{P}+(r_t-q_t)S\frac{1}{S}\partial_x\tilde{P}+\frac{1}{2}\sigma^2(t,S)S^2\frac{1}{S^2}(\partial^2_{xx}\tilde{P}-\partial_x\tilde{P})-r_t\tilde{P}=0$$

which simmplifies into

$$\begin{cases}\partial_t\tilde{P}(t,x)+\frac{1}{2}\sigma^2(t,F_te^x)(\partial^2_{xx}\tilde{P}-\partial_x\tilde{P})=0\\ \tilde{P}(T,x)=e^{-\int_0^T r_s ds}\Phi(F(0,T)e^x)\end{cases}\tag{2.3}$$

In practice, (2.3) is resolved numerically. P is obtained from $\tilde{P}$.

In the case of a eurpopean call option, the terminal condition is:

$$P(T,S)=(S-K)^+$$

$$\tilde{P}(T,x)=e^{-\int_0^T r_s ds}P(T,S)=e^{-\int_0^T r_s ds}(S-K)^+=e^{-\int_0^T r_s ds}(S_0e^{\int_0^T(r_s-q_s)ds}e^x-K)^+$$

Please note that boundary conditions have to be adapted as well.

3. Partial Differential Equation

In this section, we consider an unknown function $(t,x)\to u(t,x)$ solution of the partial differential equation of the form

$$\partial_t u+a(t,x)u+b(t,x)\partial_x u+c(t,x)\partial_{xx}u+d(t,x)=0\tag{3.1}$$

The resolution domain of such PDE is $[0,T]\times[x_m,x_M]$.
The above equation forward (resp. backward) is complemented with initial (resp. terminal) solution as well as some boundary conditions.

In finance, we usually deal with backward PDE (except for density functions where PDE are forward). The terminal condition of a payoff is then

$$u(T,x)=\psi(x);\forall x\in[\underline{x},\overline{x}] \tag{3.2}$$

The boundary conditions might by of type Dirichlet or Von Neumann. In this project, only Dirichlet type is considered:

$$\forall t\in[0,T];\begin{cases}u(t,x_m)=u_-(t)\\ u(t,x_M)=u_+(t)\end{cases} \tag{3.3}$$

Discrete problem

The resolution grid is used where the resolution domain is discretized into
$(t_j)_{0\le i\le n}$ and
$(x_j)_{0\le j\le m}$ with $t_0=0$, $t_n=T$ and $x_0=\underline{x}$, $x_m=\overline{x}$

Let $\delta t_i=t_{i+1}-t_i$ and $\delta x_j=x_{j+1}-x_j=\delta x$ (note that (x_j) is a uniform discretization)

The derivatives in the PDE equation are replaced by numerical The following numerical schemes:

- $\partial_t u(t_i,x_j)=\frac{u_{i+1,j}-u_{i,j}}{\delta t_i}$
- $\partial_x u(t_i,x_j)=\frac{u_{i,j+1}-u_{i,j-1}}{2\delta x}$
- $\partial^2_{xx}u(t_i,x_j)=\theta\frac{u_{i,j+1}+u_{i,j-1}-2u_{i,j}}{\delta x^2}+(1-\theta)\frac{u_{i+1,j+1}+u_{i+1,j-1}-2u_{i+1,j}}{\delta x^2}$

Thus, the PDE could be written as

$$\frac{u_{i+1,j}-u_{i,j}}{\delta t_i}+a_{i,j}u_{ij}+b_{i,j}\frac{u_{i,j+1}-u_{i,j-1}}{2\delta x}+c_{i,j}(\theta\frac{u_{i,j+1}+u_{i,j-1}-2u_{i,j}}{\delta x^2}+(1-\theta)\frac{u_{i+1,j+1}+u_{i+1,j-1}-2u_{i+1,j}}{\delta x^2})+d_{i,j}=0$$

Putting $u_{i,.}$ and $u_{i+1,.}$ together, one has $\forall j\in[1\dots m-1]$

$$\begin{aligned}&[(-\frac{b_{i,j}}{2\delta x}+\frac{\theta c_{i,j}}{\delta x^2})u_{i,j-1}-(\frac{1}{\delta t_i}+\frac{2\theta c_{i,j}}{\delta x^2}-a_{ij})u_{i,j}+(\frac{b_{i,j}}{2\delta x}+\frac{\theta c_{i,j}}{\delta x^2})u_{i,j+1}]\\&+[\frac{(1-\theta)c_{i,j}}{\delta x^2}u_{i+1,j-1}+(\frac{1}{\delta t_i}-\frac{2(1-\theta)c_{i,j}}{\delta x^2})u_{i+1,j}+\frac{(1-\theta)c_{i,j}}{\delta x^2}u_{i+1,j+1}]+d_{i,j}=0\end{aligned} \tag{3.4}$$

Let U_i be the vectorial representation of the solution at time t_i, that is
$U_i=(u_{i,j})_{j\in[0\dots m]}$

Let P_i and Q_i be the matrices of size $(m-1)\times(m-1)$ defined below:

$$P_i=\begin{pmatrix}a_{i,1}-(\frac{1}{\delta t_i}+\frac{2\theta c_{i,1}}{\delta x^2}) & (\frac{b_{i,1}}{2\delta x}+\frac{\theta c_{i,1}}{\delta x^2}) & 0 & \dots & 0\\ -\frac{b_{i,2}}{2\delta x}+\frac{\theta c_{i,2}}{\delta x^2} & a_{i,2}-(\frac{1}{\delta t_i}+\frac{2\theta c_{i,2}}{\delta x^2}) & \frac{b_{i,2}}{2\delta x}+\frac{\theta c_{i,2}}{\delta x^2} & \dots & 0\\ \vdots & \vdots & \ddots & \vdots & \vdots\\ 0 & \dots & 0 & -\frac{b_{i,m-1}}{2\delta x}+\frac{\theta c_{i,m-1}}{\delta x^2} & a_{i,m-1}-(\frac{1}{\delta t_i}+\frac{2\theta c_{i,m-1}}{\delta x^2})\end{pmatrix}$$

and

$$Q_i = \begin{pmatrix} \frac{1}{\delta t_i} - \frac{2(1-\theta)c_{i,1}}{\delta x^2} & \frac{(1-\theta)c_{i,1}}{\delta x^2} & 0 & \cdots & 0 \\ \frac{(1-\theta)c_{i,2}}{\delta x^2} & \frac{1}{\delta t_i} - \frac{2(1-\theta)c_{i,2}}{\delta x^2} & \frac{(1-\theta)c_{i,2}}{\delta x^2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & \cdots & 0 & \frac{(1-\theta)c_{i,m-1}}{\delta x^2} & \frac{1}{\delta t_i} - \frac{2(1-\theta)c_{i,m-1}}{\delta x^2} \end{pmatrix}$$

Finally, let V_i be the vector

$$V_i = \begin{pmatrix} d_{i,1} + (-\frac{b_{i,1}}{2\delta x} + \frac{\theta c_{i,1}}{\delta x^2})u_{i,0} + \frac{(1-\theta)c_{i,1}}{\delta x^2}u_{i+1,0} \\ d_{i,2} \\ \vdots \\ d_{i,m-2} \\ d_{i,m-1} + (\frac{b_{i,m-1}}{2\delta x} + \frac{\theta c_{i,m-1}}{\delta x^2})u_{i,m} + \frac{(1-\theta)c_{i,m-1}}{\delta x^2}u_{i+1,m} \end{pmatrix}$$

The discretized PDE (3.4) could be written as

$$P_iU_i + Q_iU_{i+1} + V_i = 0 \tag{3.5}$$

In the backward seeting , one can see that

$$U_i = -(P_i)^{-1}(Q_iU_{i+1} + V_i)$$

C++ project

Using the development above, please write a C+ program that

- defines a matrix class with function to check if the matrix is invertible and returns its inverse.
- implements a PDE pricer where the inputs are:
  - time grid as double vector of size n. For simplicity, you can as well take as input a product maturity T and a number of timesteps n. Then, your constructor has to set the time grid with constant timestep $\delta T = \frac{T}{n}$
  - space grid as a double vector of size m. Use a uniform grid with boundaries $\pm\lambda\sigma_0$ where λ is a multiplier you choose and σ_0 is equal to you at-the-money volatility - (here we will deal with constant volatilities)
  - the boundary conditions (as two vectors of double of size n)
  - the terminal condition (as a vector of double of size m)
  - the matrices given values of function a, b, c and d
- implement a function that gives the Black-Scholes price of call/put vanilla option
- in your main file,
  - value the price of a call option of a constant colatility.
  - Compare this price to the price given by closed form formula of Black-Scholes.
  - Study the convergence (decreasing the times space and the x-steps or equivalently increasing n and m)

Bonus: Do it also with Monte Carlo