Monte Carlo and Finite Difference Methods
American Options

Imad BADDA[1], Elias TOUIL[1]

June 17, 2023

Abstract

This project is part of the evaluation of the course "Monte Carlo and Finite Difference" of the MASEF (ex DEA-MASE) of the Paris Dauphine University. The objective is to use the knowledge acquired during the course to understand a corpus of scientific articles, to be able to understand and reproduce the reasoning and algorithms described there. Our choice of subject is the pricing of American options and this choice is not innocuous as it also allows us to make a link with elements seen during the course "Stochastic Control". Therefore we will in this project explain what an american option is, what is the difficulty linked to its pricing, how it can be overcame and we will even go further by studying other properties of american options. The first section deals with how it can be priced through Monte-Carlo, finding a lower and an upper bound of the price. The second sections deals with how it can be priced through PDE, numerical schemes involved, how it can be improved and finally and how we can compute greeks.

[1]master student, Mathématiques de l'Assurance de l'Economie et de la Finance (MASEF),Paris Dauphine University, 75016 Paris, France.

Contents

1 Introduction

On financial markets there exist special contracts entitled : options. Those contracts give the right (not the obligation) to the holder to exercise the contract.

These contracts can be divided in two big families, european and american options. The difference between those two is that for european options, the holder can only exercise its right at the maturity of the option whereas for american options, the holder can exercise his right at any time prior to the maturity. This simple difference leads to many difficulty in the pricing of such contracts. In this project we will see methods to price an american option by probabilistic method (and get a good confidence interval) as well as deterministic methods. A big part of the probabilistic method is due to L.C.G. Rogers as he gives a dual formulation of the pricing of the american option [6].

For the PDE part, we will focus on numerical schemes and the difference between pricing an european option, and it's american counterpart. Finally we will calculate the greeks as well, the Delta and the Gamma.

2 Pricing by Monte-Carlo

2.1 Mathematical Context

Let $T > 0$, a time horizon, and $\left(\Omega, \mathcal{F}, (\mathcal{F}_t)_{0\leqslant t\leqslant T}, \mathbb{P}\right)$ a filtered space ,(r_t) the adapted process representing the interest rate. We'll suppose the latter to be deterministic of the form : $r_t = e^{rt}$, $\forall 0 \leqslant t \leqslant T$, for a certain r>0.

Denoting $(W_t)_{0\leqslant t\leqslant T}$ a d-dimensional brownian motion and $(S_t)_{0\leqslant t\leqslant T}$, the price of the underlying (which has the same dimension as W), is the solution of the following SDE :

$$\begin{cases} dS_t &= \mathrm{S}_t \left(rdt + \sigma dW_t\right) \\ S_{t_0} &= \mathrm{x} \end{cases} \tag{Black-Scholes}$$

The adapted reward process $(Z_t)_t$, also called payoff, is generally a function of the underlying asset S_t, i.e : $\tilde{Z}_t = f(S_t)$. We'll note its discounted counterpart as follow : $Z_t = \exp\left(-\int_0^t r_s ds\right)\tilde{Z}_t$. We finally introduce the following flow notation : $S_t := S_t^{t_0,x_0}$.
An american option is an option that gives the right to the holder to excerise the option at any time prior to the maturity. Therefore this kind of options is more flexible than their european counterparts as one can exercise before the maturity. let's denote by $v(t,x)$ the price of an american option at time t and $S_t = x$:

$$v(t,x) = \underset{\tau\in\mathcal{T}_t}{essup}\left\{\ \mathbb{E}^{\mathbb{Q}}\left[Z_\tau|\mathcal{F}_t\right]\ \right\} \tag{American option}$$

Where $\mathcal{T}_t$ denotes the set of stopping times with values in $[t;T]$

2.2 Lower Bound

2.2.1 The problem

First of all, as we are dealing with numerical methods, we will have to discretize the time interval, i.e : define $(t_i)_{0\leqslant i\leqslant m}$ such that : $0 = t_0 < t_1 < ... < t_m = T$
One may remark that by doing so, we can only obtain a lower bound (i.e : suboptimal value) of the true price of an american option. Indeed as the sup is only seek in $\{t_0, t_1, .., t_m\} \subset [0,T]$, the optimizing procedure over the discrete counterpart of the interval may lead to miss stopping times providing better values of the conditional expectation.
As stated in [4] the price of an american option has the following dynamic programming equation :

$$\begin{aligned} \tilde{v}(t_m, x) &= \tilde{Z}_{t_m} \\ \tilde{v}(t_{i-1}, x) &= \max\left(\tilde{Z}_{t_i}\ ,\ \mathbb{E}\left[D_{i-1,i}\tilde{v}\left(t_i, S_{t_i}^{t_{i-1},x}\right)\Big| S_{t_{i-1}} = x\right]\right) \quad \forall i \in [1,m] \end{aligned}$$

where $D_{i-1,i} = \exp\left(-\int_{t_{i-1}}^{t_i} r_s ds\right)$.
If r is constant and the time steps are evenly spaced (i.e such that $t_i = i\frac{T}{m} = ih,\ \forall 0 \leqslant i \leqslant m$), then we can call the discount factor over one period D, and $D = e^{-rh}$. Thus, we have a backward formulation of the problem.
The dynamic programming formulation leads to consider the option pricing problem as finding the right value, and one can realize that it's the same as finding the good stopping rule (i.e a stopping time $\tau^\star$ such that $v(t,x) = \mathbb{E}\left[\tilde{Z}_{\tau^\star}\Big|\mathcal{F}_t\right]$)

A naive numerical approach of the problem using nested Monte-Carlo simulations would lead to an computationally untractable solution as the complexity of the procedure would grow exponentially with the number of time steps.

2.2.2 The Longstaff Schwartz algorithm

An algorithmic implementation of the dynamic programming equation is the Longstaff-Schwartz algorithm. To summarize, it relies on an approximation of the conditional expectations by fitting a parametrized function of the $L^2(\Omega)$ space, rather than trying approach it by extensive Monte-Carlo simulations. Estimating these expectation lead to estimate continuation value, thus an exercise policy.

Following the natural recursion flow of the equation, it allows a robust and computationally acceptable procedure. The core of the conditional expectation comes from the use of basis functions. Here, we used :

$$f_1 : x \to 1$$
$$f_2 : x \to x$$
$$f_3 : x \to x^2$$

We want to find $\alpha_1, \alpha_2, \alpha_3$ such that for f defined as :

$$f : x \to \alpha_1 + \alpha_2 x + \alpha_3 x^2$$

can be used such that $f(S_{t_k})$ is a good approximation of $\mathbb{E}\left[\tilde{v}(t_{k+1}, S_{t_{k+1}})|S_{t_k}\right]$, i.e. calibrating $(\alpha_1^k, \alpha_k^2, \alpha_k^3)_{1\leqslant k \leqslant m-1}$ by optimizing recursively over the following problem :

$$\arg \min_{\alpha_1,\alpha_2,\alpha_3 \in \mathbb{R}^3} \left(\mathbb{E}\left[v(t_{k+1}, S_{t_{k+1}})|S_{t_k}\right] - \left(\alpha_1 + \alpha_2 S_{t_k} + \alpha_3 S_{t_k}\right) \right)^2$$

We implemented and tested this method on various parameters. Here is an example of the results we obtained :

S_0	American (true)	American (LS)	IC
80	21.6059	21.4741	[21.4721,21.7175]
85	18.0374	17.7453	[17.4558,17.9798]
90	14.9187	14.5793	[14.2816,14.9826]
95	12.2314	11.7301	[11.6175,12.1861]
100	9.9458	9.4875	[9.3507,9.9187]
105	8.0281	7.4204	[7.5066,7.9311]
110	6.4352	5.93186	[5.8539,6.1331]
115	5.1265	4.5668	[4.6054,4.8294]
120	4.0611	3.5391	[3.6417,3.8476]

Table 1: LS Pricing of a Standard American Put $|K = 100$, $r = 0.06$, $T = 0.5$, $\sigma = 0.4$ | $m = 100$,$n_{sim} = 5000$

As stated previously, this methods gives lower bounds of the price.

2.3 Upper Bound

2.3.1 The dual problem

The methods seen earlier can only produce a lower bound. It can be an issue, because even if the the method converges toward the real price (if the time step goes to 0 and the number and sample paths goes to $+\infty$) it still converges by below, which may be no suitable if we want a proper confidence interval for the **true price**. Therefore we need to consider other methods.

First of all, one can see that what we are dealing with is an optimisation problem. Indeed the problem considered can be seen as the primal. One could thus think of the dual problem. The dual problem is the following :

$$v(t,x) = \underset{M \in H_0^1}{ess\inf} \left(\mathbb{E}^{\mathbb{Q}} \left[\sup_{0 \leqslant s \leqslant T} (Z_s - M_s) \Big| \mathcal{F}_t \right] \right) \tag{Dual problem}$$

Where H_0^1 is the following set :

$$H_0^1 = \{M \text{ martingale s.t }, \sup(M_t) < +\infty \text{ and } M_0 = 0\}$$

2.3.2 Constructing the good martingale

A dual problem means a dual space (in the sense of optimization). Thus, the initial problem has been transformed into a variational problem on a space of martingales. The two papers we studied elaborated two developments to tackle this challenge.

Rogers method [6] While exhibiting heuristic over the martingale choice procedure by interpretating various inequalities derived from the problem, Rogers tries out to perform pricing with a crude choice of martingale : the corresponding European put option.

S_0	American (true)	American (Rogers)	IC
80	21.6059	21.7874	[21.7839,21.7917]
85	18.0374	18.1626	[18.1569,18.1641]
90	14.9187	14.9963	[14.9949,15.0024]
95	12.2314	12.2832	[12.2798,12.2857]
100	9.9458	9.9791	[9.9785,9.9843]
105	8.0281	8.0483	[8.0459,8.0506]
110	6.4352	6.4473	[6.4455,6.4496]
115	5.1265	5.1327	[5.1312,5.1346]
120	4.0611	4.0639	[4.0635,4.0664]

Table 2: Rogers estimates | $K = 100$, $r = 0.06$, $T = 0.5$, $\sigma = 0.4$ | $m = 100$, $n_{sim} = 5000$

As stated, the Rogers estimates give an upper bound of the prices.

Andersen-Broadie [1] The article exhibit a method allowing to build sequentially a martingale, exploiting an exercise region provided by a primal algorithm. In particular, it uses the following relation, allowing to extract the martingale part M of a process X :

$$M_{k+1} = M_k + X_k - \mathbb{E}[X_k \mid \mathcal{F}_{k-1}]$$

Unfortunately, even if we coded the method, we could not make it work properly.
In any case, we expected it to provide a better quality upper bound as the primal algorithm correctly approximated the exercise region. We tried to use Longstaff-Schwartz for this, by expressing the regression coefficients it determines at each time step.

3 Pricing by PDE

3.1 Understanding the problem

3.1.1 Context and variational inequality

The price of an American option is determined by solving a stochastic control problem, specifically an optimal stopping problem. This is because the control of the primal problem is the the optimal stopping policy.

$$
\begin{aligned}
v(t,x) &= \underset{\tau \in \mathcal{T}_t}{essup} \left(\mathbb{E}^{\mathbb{Q}} \left[Z_\tau | \mathcal{F}_t \right] \right) && (1) \\
&= \underset{\tau \in \mathcal{T}_t}{essup} \left(\mathbb{E}^{\mathbb{Q}} \left[e^{-\int_t^\tau r_u du} g(S_\tau) \middle| \mathcal{F}_t \right] \right) && (2)
\end{aligned}
$$

So as stated in [5], v is solution to the following problem :

$$
\max \left(\frac{\partial v}{\partial t}(t,x) + rx\frac{\partial v}{\partial x}(t,x) + \frac{1}{2}\sigma^2 x^2 \frac{\partial v}{\partial x^2}(t,x) - rv(t,x), g(x) - v(t,x) \right) = 0 \tag{3}
$$

Still in the same reference, we can see that solving numerically this problem is the same as the pricing of an european option with just one extra step : taking the maximum between two values. We will therefore concentrate on the PDE and add the max between the payoff and the value of the PDE. Therefore we can just say that we solve the european version of the option and just add our condition to make it american.

Another interesting point is to charaterize when the solution v is either found by the PDE or either the payoff, i.e : "Should the option be exercised or should I wait ?". This gives rise to two regions, the continuation and the stopping region which are respectively defined as :

$$
\begin{aligned}
C &= \{(t,x), \in [0,T[\times \mathbb{R}_+^*, v(t,x) > g(x)\} && (4) \\
D &= \{(t,x), \in [0,T[\times \mathbb{R}_+^*, v(t,x) = g(x)\} && (5)
\end{aligned}
$$

3.1.2 Well posed PDE

As we saw earlier, the PDE part of the equation only makes sense in the continuation region and it corresponds to the pricing of an european put. Therefore we will focus on the pricing of an european put and then we will add the condition that transform the european into an US option.

For our problem to be well-posed, we need to define boundary conditions. As we are pricing an put option, the following conditions are natural to use :

$$
\begin{aligned}
V(t,0) &= Ke^{-r(T-t)} \\
V(t,x) &\underset{x \to +\infty}{\longrightarrow} 0
\end{aligned}
$$

Therefore we have the following problem :

$$
\begin{cases}
\frac{\partial v}{\partial t}(t,x) + rx\frac{\partial v}{\partial x}(t,x) + \frac{1}{2}\sigma^2 x^2 \frac{\partial v}{\partial x^2}(t,x) - rv(t,x) = 0 \quad (t,x) \in]0,T[\times \mathbb{R}_+^* \\
v(T,x) = (K-x)^+ \quad \forall x \in \mathbb{R}^+ \\
v(t,0) = Ke^{-r(T-t)} \quad \forall t \in [0,T] \\
\lim\limits_{x \to +\infty} v(t,x) = 0 \quad \forall t \in [0,T]
\end{cases}
$$

3.2 Numerical Methods

3.2.1 Choosing the discretization and bounded domain

As we don't have analytical solutions, we will have to rely on numerical solution. So we can't consider all values x and all times t so we need to consider a time grid and a price grid. We will respective denote them by : $(t_j)_{0 \leqslant j \leqslant m}$ and $(x_i)_{0 \leqslant i \leqslant n}$ which are defined as follow :

$$t_j = t_{min} + j\frac{t_{max} - t_{min}}{m} = t_{min} + j\tau$$
$$x_i = x_{min} + i\frac{x_{max} - x_{min}}{n} = x_{min} + ih$$

Here t_{min} and t_{max} are obvious and are respectively : 0 and T. For x_{min} and x_{max} the choice depends more on the problem faced therefore they will be precised when dealing with one specific problem. Here we are dealing with a put option. Therefore we can as well find x_{min} and x_{max} quite easily as we only need to take $x_{min} = 0$ as the price of an asset can't be negative and we only need to take x_{max} big enough to make sense. By big enough we mean a price that is "realistically" unattainable by the asset in the time frame considered. We will go by a multiple of the strike which could be for example : $x_{max} = 4K$

3.2.2 Implicit Scheme

We will use an implicit scheme as it will be better for stability stuff as we don't need to check a CFL.

Considering the derivatives, we will be using the following approximations :

$$\begin{aligned}
\frac{\partial v}{\partial x}(t_j, x_i) &= \frac{v(t_j, x_{i+1}) - v(t_j, x_{i-1})}{2h} + \mathcal{O}(h^2) \approx \frac{v(t_j, x_{i+1}) - v(t_j, x_{i-1})}{2h} \\
\frac{\partial v}{\partial t}(t_j, x_i) &= \frac{v(t_{j+1}, x_i) - v(t_j, x_i)}{\tau} + \mathcal{O}(\tau) \approx \frac{v(t_{j+1}, x_i) - v(t_j, x_i)}{\tau} \\
\frac{\partial^2 v}{\partial x^2}(t_j, x_i) &= \frac{v(t_j, x_{i-1}) - 2v(t_j, x_i) + v(t_j, x_{i+1})}{h^2} + \mathcal{O}(h^2) \approx \frac{v(t_j, x_{i-1}) - 2v(t_j, x_i) + v(t_j, x_{i+1})}{h^2}
\end{aligned}$$

With such approximation, we will end up with a numerical scheme with error in $\mathcal{O}(\tau)$ and $\mathcal{O}(h^2)$. In order to have a more concise and compact notation, we will denote v_i^j by $v(t_j, x_i)$ and thus if we look at the PDE, we get :

$$\begin{aligned}
\frac{v_i^{j+1} - v_i^j}{\tau} + rx_i\frac{v_{i+1}^j - v_{i-1}^j}{2h} + \frac{1}{2}\sigma^2 x_i^2 \frac{v_{i-1}^j - 2v_i^j + v_{i+1}^j}{h^2} - rv_i^j &= 0 && \forall (i,j) \in [1, n-1] \times [1, m-1] \\
v(t_m, x_i) &= (K - x_i)^+ && \forall i \in [0, n] \\
v(t_j, x_0) &= Ke^{-r(T-t)} && \forall j \in [0, m] \\
v(t, j, x_n) &= 0 && \forall j \in [0, m]
\end{aligned}$$

One can remark that the first equation can be rewritten as :

$$\begin{aligned}
v_i^{j+1} &= v_{i-1}^j \left[\frac{\tau r x_i}{2h} - \frac{\sigma^2 x_i^2 \tau}{2h^2}\right] + v_i^j \left[\frac{\sigma^2 x_i^2 \tau}{h^2} + \tau r + 1\right] + v_{i+1}^j \left[-\frac{\tau r x_i}{2h} - \frac{\sigma^2 x_i^2 \tau}{2h^2}\right] \\
&= a_i v_{i-1}^j + b_i v_i^j + c_i v_{i+1}^j
\end{aligned}$$

First, let's remark that each coefficient a_i, b_i, c_i is independant of the time in the sense that the only part where the time plays a role is the time step τ but as we decided to take it as constant, at each time if we wanted to calculated the coefficient - a_i, b_i, c_i - they would be the same.

Let's denote by V^j the following vector : $V^j = (v_1^j, ..., v_{n-1}^j)$ Therefore one can see the equations as the following system :

$$V^{j+1} = AV^j + B^j$$

Where

$$A = \begin{pmatrix}
b_1 & c_1 & 0 & \cdots & 0 \\
a_2 & b_2 & c_2 & 0 & 0 \\
0 & \ddots & \ddots & \ddots & 0 \\
\vdots & 0 & a_{n-2} & b_{n-2} & c_{n-2} \\
0 & 0 & 0 & a_{n-1} & b_{n-1}
\end{pmatrix}$$

and

$$B^j = \begin{bmatrix}
a_1 v_0^j \\
0 \\
\vdots \\
c_{n-1} v_n^j
\end{bmatrix}$$

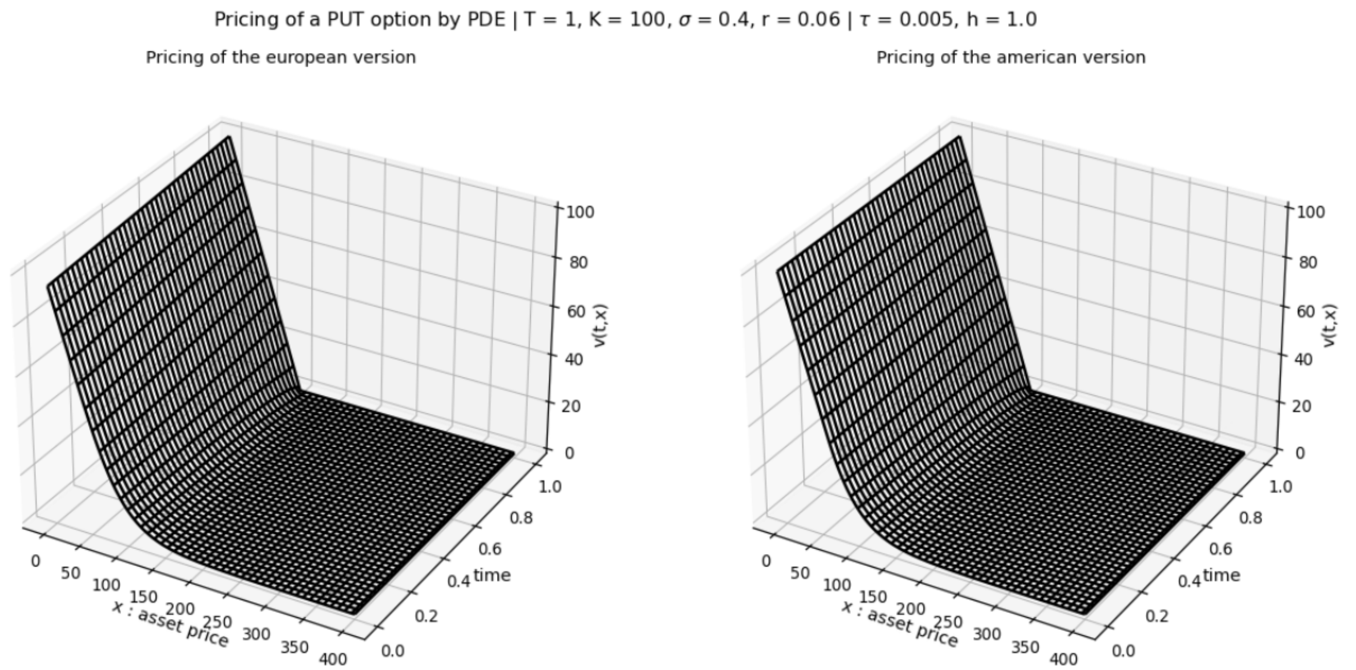
And as we have the terminal condition, at time t_j, the unknown is V^j. So we are trying to solve a linear system of the form :

$$AX = b$$

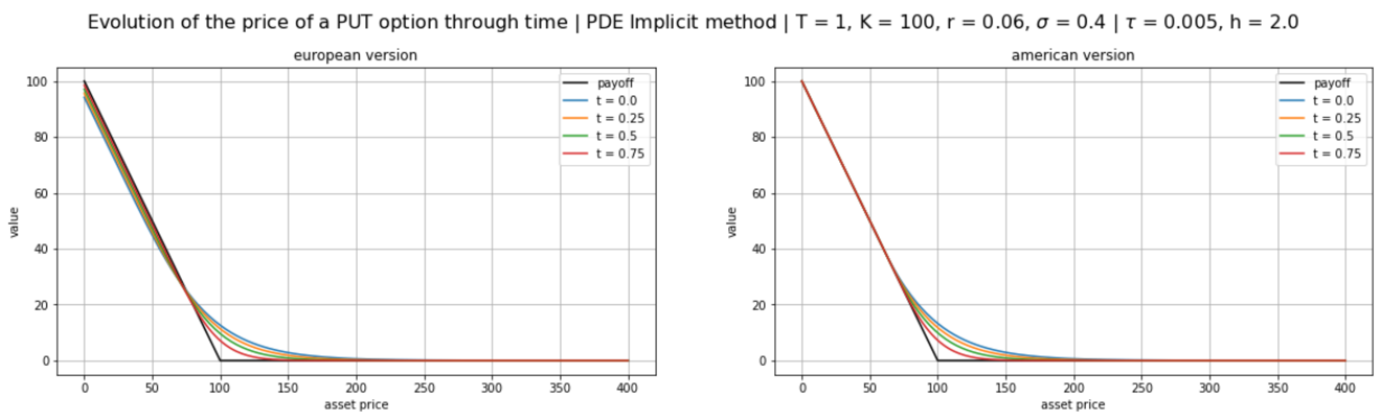
We will first do it by inversing the matrix A but later we will go further with different techniques which are way better to solve a linear system.

3.2.3 Pricing of an American Put

The american put is priced by the splitting method where we compute v as the solution of the PDE and then apply the max between this value and the payoff.



The evolution of our european put option seems very alike what we can find in the litterature. Our code seems to work great



We will look at the price we have by comparing them to the "true price" given in [6].

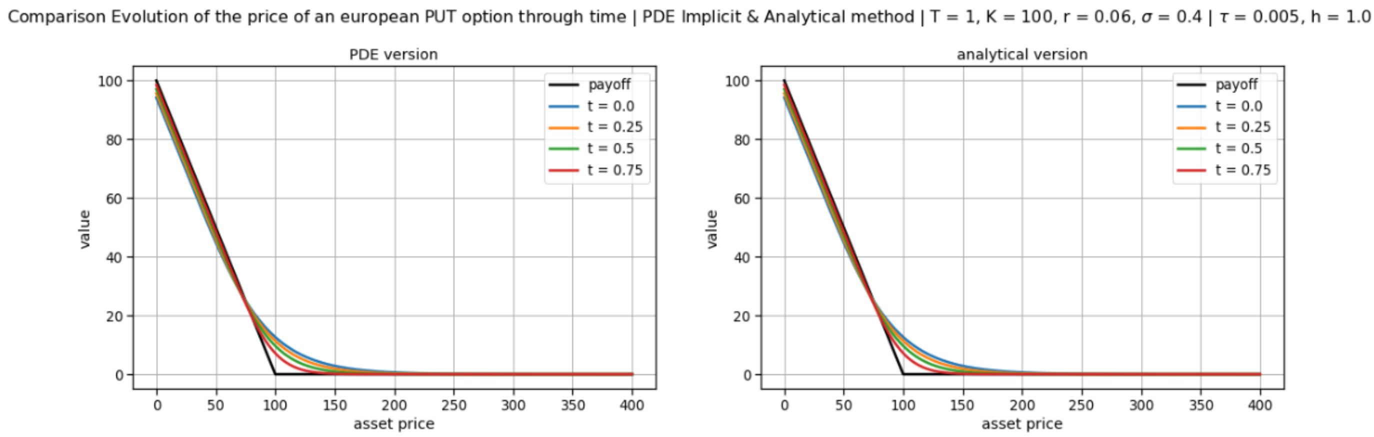
Table 3: Pricing of a Standard American Put | $K = 100$, $r = 0.06$, $T = 0.5$, $\sigma = 0.4$ | $m = 200$, $n = 400$

S0	American (true)	American (PDE)	error
80	21.6059	21.593367	0.012533
85	18.0374	18.022612	0.014788
90	14.9187	14.902374	0.016326
95	12.2314	12.214351	0.017049
100	9.9458	9.929712	0.016088
105	8.0281	8.012113	0.015987
110	6.4352	6.420956	0.014244
115	5.1265	5.114325	0.012175
120	4.0611	4.051312	0.009788

As expected, the values obtained by the finite difference method are close to the true values.

3.2.4 Error of the scheme on the PDE

Earlier we stated that the scheme is supposed to be in in $\mathcal{O}(\tau)$ and $\mathcal{O}(h^2)$. We could just stop there as it's the theoretical convergence. However something interesting to consider is that by computing the price of an american option, we can as well compute the price of it's european counterpart. Furthermore we've got analytical solutions of the price of an european put. Therefore we can study the error of our scheme on the european option to "validate" it. If it's good for this, it should be good for the american version.



We will look at two different errors :

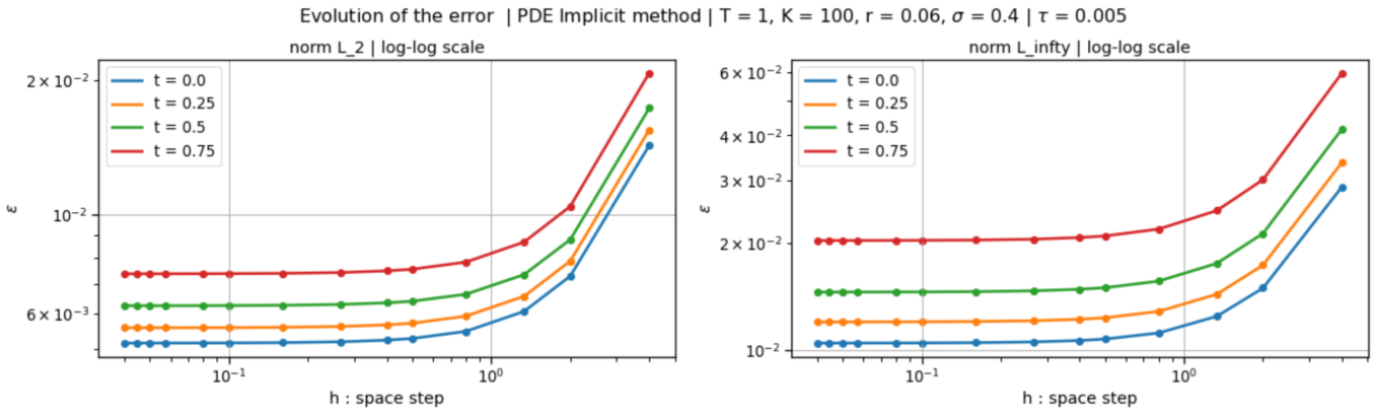
- the ℓ_2 error :

$$\varepsilon_2 = \sqrt{\frac{1}{n}\sum_{i=0}^{n}(u_i - \tilde{u}_i)^2}$$

- the ℓ_∞ error :

$$\varepsilon_\infty = \sup_{i\in[0,n]}(|u_i - \tilde{u}_i|)$$

where $\tilde{u}$ denotes the exact solution of the european PUT option and u_i the solution we got by finite difference.



As we can see the error decreases as the space step reduce. However it seems to stagnate but it's normal as the global error is linked to the size of the time step. The scheme is $\mathcal{O}(\tau)$ and $\mathcal{O}(h^2)$ therefore when one of the time step become significantly lower than the other, the other is dominant and drives the global error of the scheme.

3.2.5 Propositions for a better solver

The numerical methods that were used are good, coherent results were obtained within reasonable computing time. However they can easily be improved. One can improve the speed convergence as well as the computing time.

3.2.5.1 Richardson extrapolation The implicit scheme used earlier is stable in norm L_2 and L_∞. It does not suffer from spurious oscillations as the Cranck-Nicholson scheme [3]. However it has a big issue, the convergence in time is of $\mathcal{O}(\tau)$ whereas CN has a convergence in time of $\mathcal{O}(\tau^2)$. One way to overcome this issue is to apply the Richardson extrapolation in time.

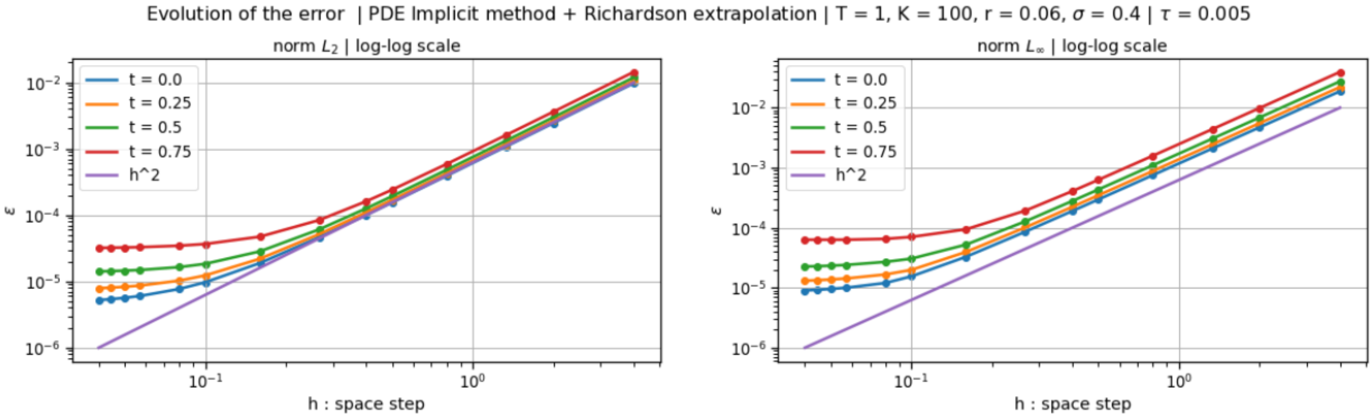
$$
\begin{aligned}
v(t_j+\tau,x_i) &= v(t_j,x_i) + \tau\frac{\partial v}{\partial t}(t_j,x_i) + \frac{\tau^2}{2!}\frac{\partial^2 v}{\partial t^2}(t_j,x_i) + \frac{\tau^3}{3!}\frac{\partial^3 v}{\partial t^3} + \mathcal{O}(\tau^3)\\
\Leftrightarrow \frac{v(t_j+\tau,x_i) - v(t_j,x_i)}{\tau} &= \frac{\partial v}{\partial t}(t_j,x_i) + \frac{\tau}{2!}\frac{\partial^2 v}{\partial t^2}(t_j,x_i) + \frac{\tau^2}{3!}\frac{\partial^3 v}{\partial t^3} + \mathcal{O}(\tau^2)
\end{aligned}
$$

$$
\begin{aligned}
v(t_j+\frac{\tau}{2},x_i) &= v(t_j,x_i) + \frac{\tau}{2}\frac{\partial v}{\partial t}(t_j,x_i) + \left(\frac{\tau}{2}\right)^2\frac{1}{2!}\frac{\partial^2 v}{\partial t^2}(t_j,x_i) + \left(\frac{\tau}{2}\right)^3\frac{1}{3!}\frac{\partial^3 v}{\partial t^3} + \mathcal{O}(\tau^3)\\
\Leftrightarrow \frac{v(t_j+\frac{\tau}{2},x_i) - v(t_j,x_i)}{\frac{\tau}{2}} &= \frac{\partial v}{\partial t}(t_j,x_i) + \frac{\tau}{4}\frac{\partial^2 v}{\partial t^2}(t_j,x_i) + \left(\frac{\tau}{2}\right)^2\frac{1}{3!}\frac{\partial^3 v}{\partial t^3} + \mathcal{O}(\tau^2)
\end{aligned}
$$

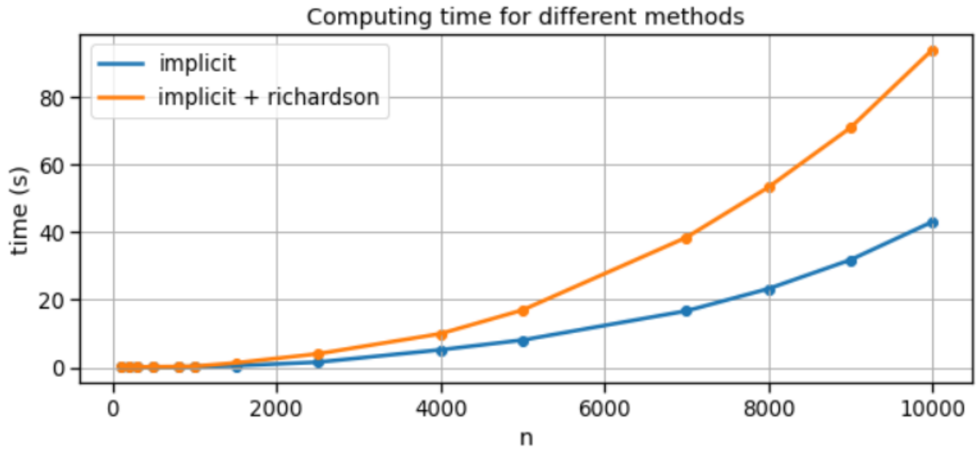
One can remark that :

$$
\frac{\partial v}{\partial t}(t_j,x_i) = 2\frac{v(t_j+\frac{\tau}{2},x_i) - v(t_j,x_i)}{\frac{\tau}{2}} - \frac{v(t_j+\tau,x_i) - v(t_j,x_i)}{\tau} + \mathcal{O}(\tau^2) \tag{6}
$$

Therefore solving the PDE with both time steps and combining them can give us a better time approximation. So if we call $v^{(1)}$ and $v^{(2)}$ the two solution with the time step τ and $\frac{\tau}{2}$. Then $v^{(3)} = 2v^{(2)} - v^{(1)}$ is a solution of the PDE and it's discretizated version such that $v_i^{(3),j} = 2v_i^{(2),2j} - v_i^{(1),j}$ has a convergence of order $\mathcal{O}(\tau^2)$ and $\mathcal{O}(h^2)$.



As we can see, the error is largely inferior to what we had with just the implicit method. It's nearly quadratic. The only thing that stop it from being quadratic is the fact that the error is related to the discretization in space. Indeed when we posed the problem, we had to put ourselves in a grid. However one of the boundary condition is a limit. Therefore we introduced an error at this point and it drives the global error of the scheme. Something we must as well take into account when we compare such methods is the time it takes to compute. Indeed we have a better convergence but how much does it change from a time consuming point of view. Well, let's take a look at it.

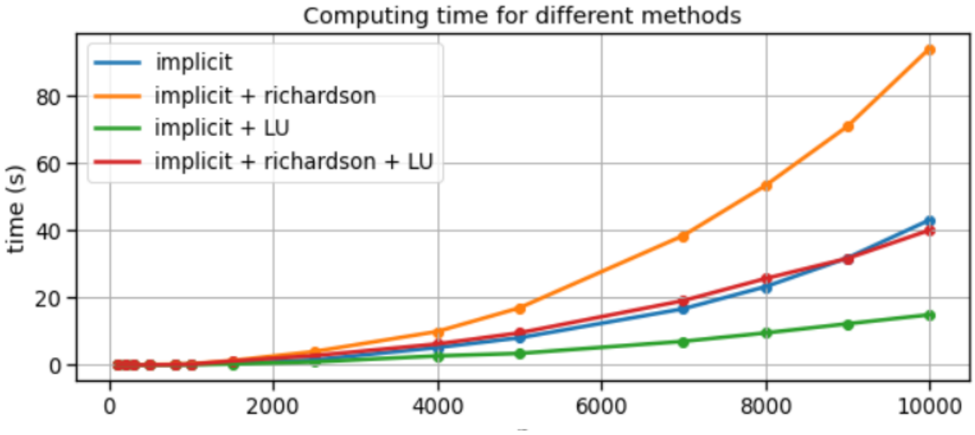


3.2.5.2 Solving linear system, a direct method : LU Decomposition As stated earlier, when solving the finite difference scheme, we solve a linear system of the form : $AX = b$. We did it by inverting the matrix A. However in most cases, inverting a matrix is a bad idea has inverting a matrix necessit itself to solve a linear system and we might loose some nice properties. Indeed in our case the matrix A is tridiagonal but A^{-1} doesn't have any reason to be tridiagonal as well. Therefore we are basically solving a linear system to have another - harder - linear system to solve.

One of the way to overcome this issue is by finding the LU decomposition [2]. Indeed the LU decomposition make it easier to solve the linear system as : $A = LU$ and thus : we just have to solve two easy system :

$$Ly = b$$
$$UX = y$$

Furthermore as A is tridiagonal, its matrix decomposition has a simple form with only two diagonals for each matrix L and U.



As we can see, the simple fact of using the LU decomposition instead of inverting the matrix gives great results in term of computing time. The computing time has been divided by two, making the Richardson extrapolation slightly more onerous than just the implicit method and therefore making it usable.

Finally we can take a look at the error we have for American PUT with the values given by Roger in [6].

Table 4: Pricing of a Standard American Put | $K = 100$, $r = 0.06$, $T = 0.5$, $\sigma = 0.4$ | $m = 200$, $n = 400$

S0	American (true)	American (PDE)	American (PDE+richardson)	error (PDE)	error (PDE+richardson)
80	21.6059	21.593367	21.603865	0.012533	0.002035
85	18.0374	18.022612	18.034525	0.014788	0.002875
90	14.9187	14.902374	14.915291	0.016326	0.003409
95	12.2314	12.214351	12.227669	0.017049	0.003731
100	9.9458	9.929712	9.942759	0.016088	0.003041
105	8.0281	8.012113	8.024267	0.015987	0.003833
110	6.4352	6.420956	6.431724	0.014244	0.003476
115	5.1265	5.114325	5.123379	0.012175	0.003121
120	4.0611	4.051312	4.058502	0.009788	0.002598

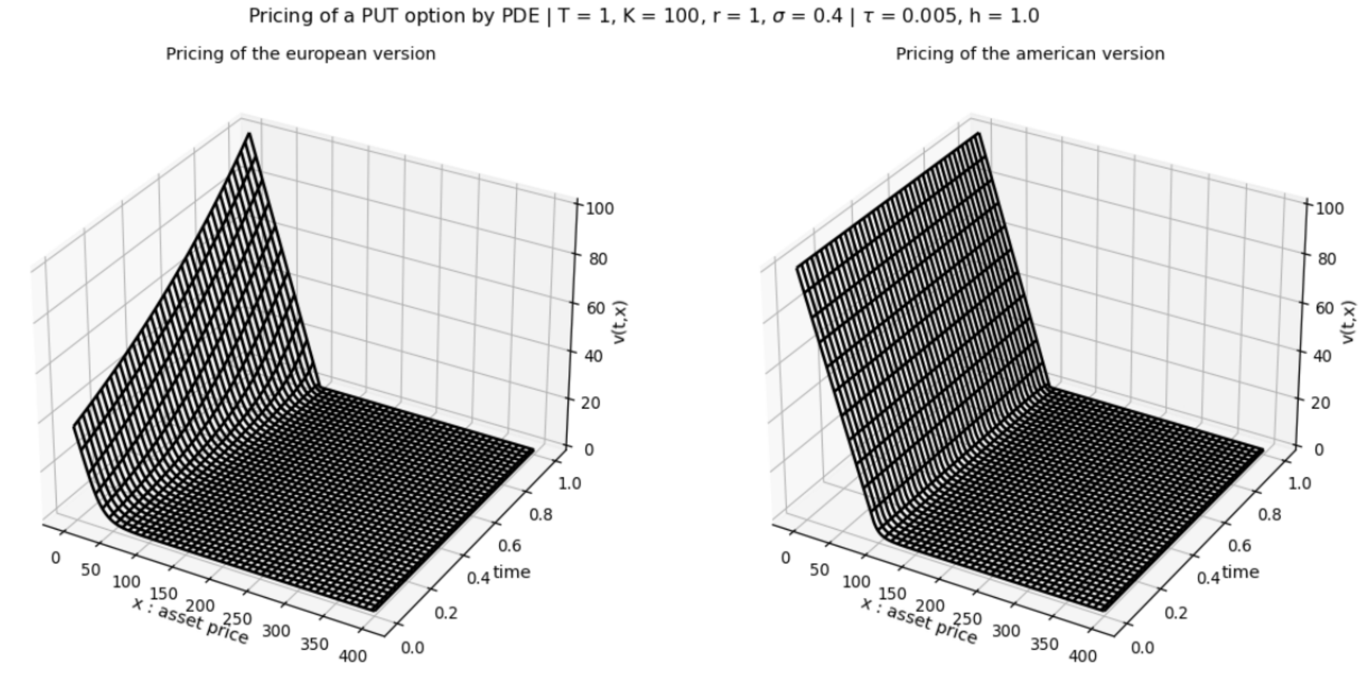
So if we want to summarize our numerical solver so far, we use an implicit scheme to get ride of the possible spurious oscillations, we use the Richardson extrapolation to have a better order of convergence and finally we solve the linear system by using the LU decomposition to solve it in a simpler / faster way.

3.3 Playing with parameters

Now that we have a solver which works, we can look at how the price of american option is influenced by its parameters. Two important parameters are to consider : interest rate and volatility.

3.3.1 Evolution of the price with different interest rates : european vs US put option

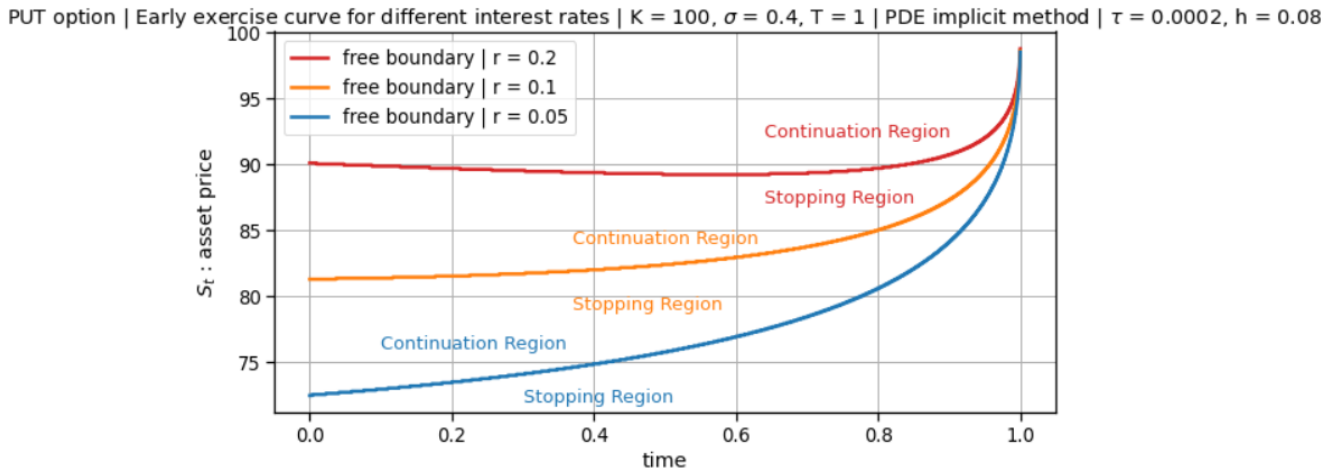
It could be interesting to look at the impact of the interest rate on the price of a put option. Indeed the interest rate plays a role in the drift of the asset. Therefore a high interest rate should add a strong deterministic, positive and increasing with time tendency to the asset price and therefore make the price of the put option lower.



As expected, the price of the european version is quite low. However for its US counterpart it's not low at all. Indeed with a high interest rate, it seems more interesting for an investor to early exercise his right and therefore the price is very often equal to the payoff. The only times it doesn't is near the strike from above as the price can't be the payoff because there is a chance for the asset to still end below the strike and therefore the option is still worth some grand.

3.3.2 Continuation and Stopping region

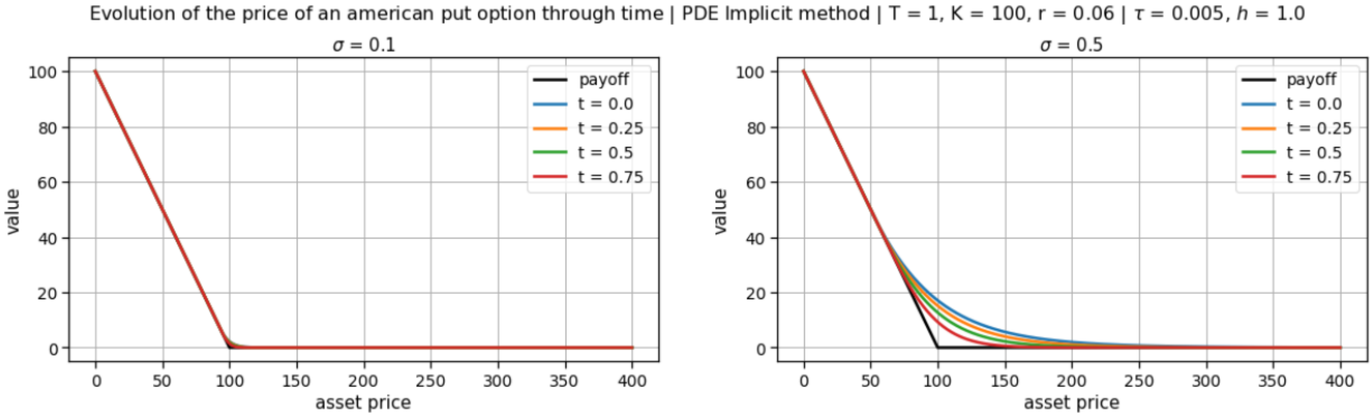
Earlier we defined what we called the continuation and the stopping region. We can as well define a so called free boundary which is the curve which split in two our range of price during time such that if we are below the curve, it's optimal to stop and if we are above the curve, it's optimal to continue (i.e : not early exercise). Such free boundary can be found by each $t \in [0, T]$, finding the price S_t such that : $V(t, S_t) < (K - S_t)^+$.



One can remark that the free boundary is highly dependent on the interest rate. Indeed if the shape is overall the same, the interest rate plays à huge part in the exercise value in at time t=0.

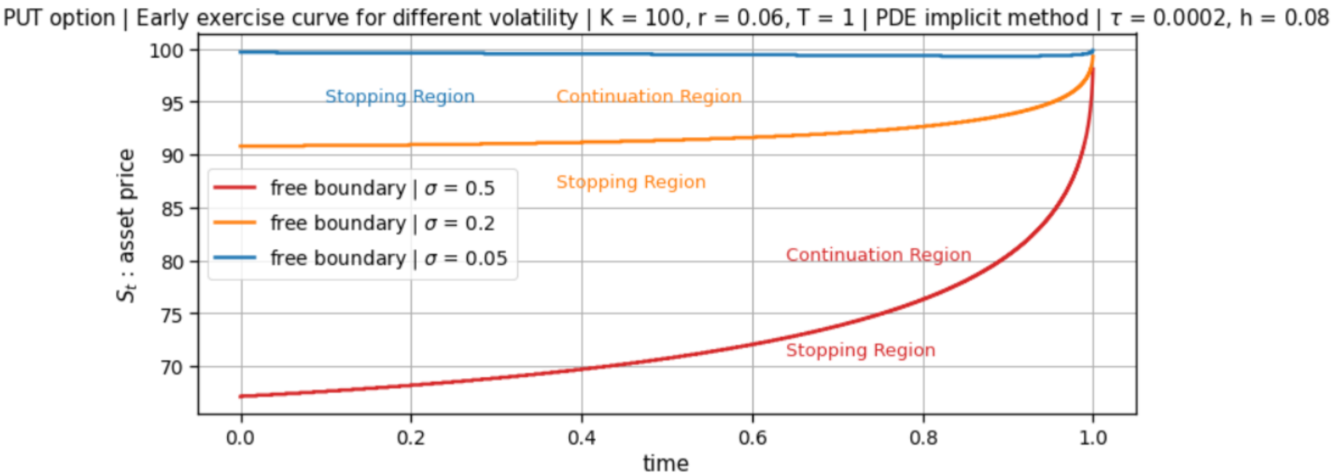
3.3.3 Evolution of the price with different volatilities

Volatility is an important parameter in the pricing of an American option because it affects the likelihood of the option being exercised. Higher volatility means that the underlying asset's price is more likely to move in a favorable direction for the option holder, making it more likely that the option will be exercised. As a result, options with higher volatility tend to be more expensive than those with lower volatility.



As expected, the price of an american option with a lower volatility is most likely to be equal to the payoff as one would early exercise nearly everytime because the price is not likely to move in a "good" direction (e.g : decrease).

3.3.4 Continuation and Stopping region



The shape of the free boundary is dependant of the volatility. Indeed the lower the volatility is, the stepper the free boundary is.

3.4 Greeks

Whenever a bank trades a derivative product, it ends up with a position that has various sources of risks. Therefore to be able to hedge it, the bank need to understand where the risk lies and how it evolves.

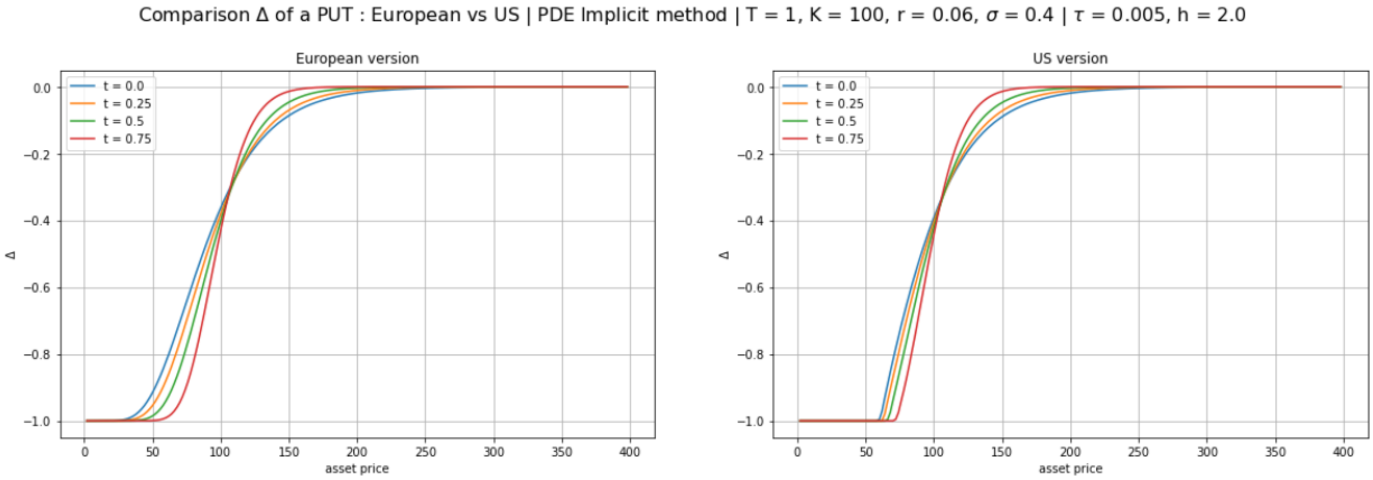
To address this problematic, we will need to know the sensitivity of the American put to the market parameters. These sensitivities are commonly referred to as the Greeks. We will look at the two most important Greeks: the Delta and the Gamma which in the context of finance are respectively, concerning the delta, the quantity of asset one has to hold in a portfolio to perfectly replicate the derivative product and, for the Gamma, the speed at which one should re-balance his portfolio.

3.4.1 Delta

Δ is the first derivative of the value of the option with respect to the underlying :it's therefore the sensibility of the price with respect to the underlying. In mathematical finance, it is the quantity of risky asset one has to have in a self-financing portfolio to perfectly replicate the option.

We will approximate it with the following formula :

$$\Delta(t_j, x_i) = \frac{v_{i-1}^j - v_{i+1}^j}{2h} \tag{7}$$



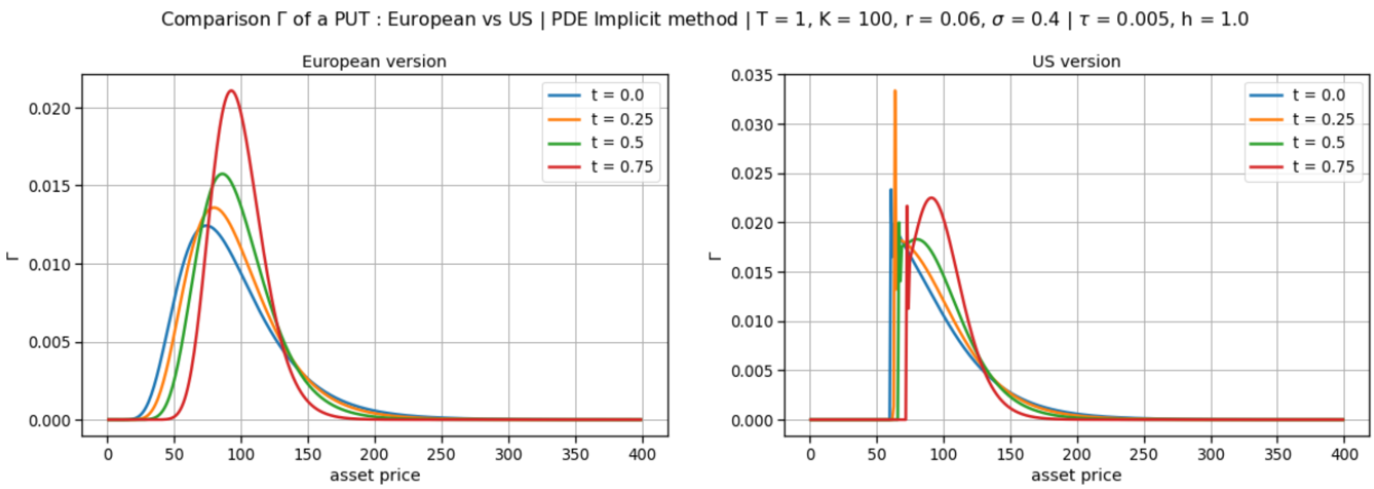
As one can see, the Delta of an european and american PUT option are different. Indeed when we are near the strike, there is no reason to early exercise because the asset price could still go lower and therefore it acts like it's european counterpart. However when we are further away from the below's strike, the holder of the option will early exercise and therefore the delta is constant (because the option acts just like the underlying).

3.4.2 Gamma

Γ is the second derivative of the value of the option with respect to the underlying. It's aswell the first derivative of the delta with respect to the underlying. Therefore it's the sensibility of the Δ with respect to the underlying. As the Δ is the quantity of risky asset we need to have to perfectly replicate the option, the Γ gives us the speed at which we should re-balance our portfolio.

We will approximate it with the following formula :

$$\Gamma(t_j, x_i) = \frac{v_{i-1}^j - 2v_i^j + v_{i+1}^j}{h^2} \tag{8}$$



As one can see, the Gamma of an european and american PUT option are different. Indeed due to the fact that an american option can be early exercised, there is a strong variation of the Gamma around the strike because if we are far away from below, the holder early exercise but as we get closer the the strike, the US option acts like the european counterpart and therefore the gamma goes from 0 to the value of the european gamma. In fact it may be a discontinuitiy.

4 Conclusion

Through this project we studied the pricing of american options with the example of the put. We explored different ways to do so.
Firstly a probabilistic approach with Monte-Carlo where we used a regression based method on conditional expectation, the longstaff schwartz algorithm which gave us a lower bound of the price. Then we used a more 'advanced' method, Roger's dual to which gave us an upper bound of the price.
Secondly we used a deterministic approach where we solved numerically the variational inequality of the put option. It has been solved by the finite difference method and we used two methods to speed up the process, the Richardson extrapolation to get a good order of convergence in time and the LU decomposition to reduce the computing time. As using the finite difference method gave us a grid of price, we explored the data to get the continuation and stopping region as well as the most common greeks : the delta and the gamma.
If many things have been done, there is still room for improvement :

- Aspects to improve :
  - (Monte-Carlo) **Variance Reduction methods**
    The use of variance reduction methods could greatly improve the computation time (hence the precision of the results for a given amount of computation ressources) for the pricing of american options.
  - (PDE) **sparse matrix**
    The use of sparse matrix could be a great way to improve the existing code as when solving the linear system, we use a $n \times n$ matrix which for big n takes a huge memory space whereas it's a tridiagional matrix. Therefore we could reduce the memory space taken. Indeed it would reduce the space taken from $\mathcal{O}(n^2)$ to $\mathcal{O}(n)$, using storage formats such as CSR.
  - **use of Cython**
    Another way to speed up the computation is to use Cython. Indeed as python is an interpreted language, it's slower than low level language such as the C language. Cython is a efficient way to reduce the gap between those two.
- Aspects to explore :
  - **Pricing by tree**
    One way of pricing american option has not been explored at all, it's the pricing by tree method (binomial and trinomial tree).
  - (PDE) **different boundary conditions**
    In our formulation of the discretized problem, we introduced Dirichlet boundaries. However by doing so as a condition is a limit, by putting a Dirichlet boundary, we are introducing an error. An idea could be to put a Neumann condition (taking the first or second derivative).
  - (PDE) **Tridiagonal Matrix Algorithm (TDMA) / Thomas algorithm**
    An algorithm specific to solve linear system of the form $AX = b$ where A is tridiagonal.
  - (PDE) **Projected Successive Over Relaxation (PSOR) algorithm**
    Iterative method based on a matrix decomposition ($A = D + L + U$) and a relaxation parameter. It should be more adapted to solve a variational inequality as the maximum is taking at every step of the iteration.
  - (PDE) **Semi-smooth Newton's method**
    Iterative method which should works better for american options.
  - (PDE) **Brennan-Schwarz Algorithm**
    Algorithm used to price Put. One issue is linked to some monotonicity condition which makes it less reliable for "general" pricing than PSOR.
  - (Monte-Carlo & PDE) **Pricing of other products**
    So far during the whole project we only priced a standard put option. It could be interesting to look at other options (digital, lookback, asian, put on an index etc.).

References

[1] L. Andersen and M. Broadie, *Primal-dual simulation algorithm for pricing multidimensional american options*, Management Science, (2004).

[2] F. Cuvelier, *Analyse numérique élémentaire.* 2022.

[3] J. Duffy. Daniel, *Numerical Methods in Computational Finance*, Wiley, 2022.

[4] P. Glasserman, *Monte-Carlo Methods In Financial Engineering*, Springer, 2003.

[5] P. Labordère, *Non Linear Option Pricing*, Chapman and Hall/CRC, 2013.

[6] L.C.G. Roger, *Monte carlo valuation of american options*, Mathematical Finance, (2002).