Two asset basket barrier option pricing with odor finite element method

Mathematical models of finance, a.y. 2019-2020

Dipartimento di Scienze Matematiche Fisiche e Informatiche
Università degli Studi di Parma

Author:
Ariel S. Boiardi *

Advisor:
Prof. Chiara Guardasoni

1 Introduction

In modern finance, assets such as stocks, bonds and commodities are often not traded directly, but used as a basis to define more complex financial products known as *derivatives*. In the last decades derivatives have become increasingly important in all branches of finance theory and practice, and consequently the problem of pricing them. A derivative can be defined as a financial product whose value depends on the values of other simpler *underlying* variables, very often the price of other traded assets.

Options, in their plain *vanilla* form, are one of the simplest forms of derivatives. Option pricing has been studied, in some form, ever since the end of the Nineteenth century [9]. In 1973 Fisher Black and Myron Scholes [4] provided a mathematical model for the evolution of the price of a vanilla option as a pure diffusion process, which enabled to price option with an analytical formula. Since then much work has been done in the field of option pricing, and more complex options have been developed.

In this paper we focus on a particular type of option, depending on a basket of two underlying assets, but whose value becomes zero if the price of the assets ever crosses some *barrier* values; for this reason this kind of options are called *barrier options*. The problem of double-barrier has been solved in analytical form by Kunitomo and Ikeda [8] for options written on one asset; the problem for basket options is still open, and therefore numerical methods are the only feasible way to price this kind of options.

Since the domain for the our pricing problem in not rectangular, we opted for the finite elements method (FEM), instead of a simpler finite difference scheme, in order to have

*Email address: arielsurya.boiardi@studenti.unipr.it

more flexibility with the discretization of the geometry.

The structure of the paper is the following. In section 2 of this paper we will introduce options in an exclusively financial framework, giving also examples of their use in financial engineering. Section 3 is devoted to developing the basic mathematical modelling of options, which is then expanded and adapted to our model problem of a two-asset double barrier option in section 4. Section 4 also contains the abstract part of the approximation procedure, which is specified to the finite element method in section 5; along with the formal description of FEM, some implementation aspects are also discussed. Finally in section 6 some numerical results obtained with our implementation are discussed. Unfortunately, since a closed-for solution of the problem does not exists and not much data was found in general literature, we are not able to benchmark our codes. All the computational work was carried out in MATLAB and codes are available in the following repository: https://github.com/arielsboiardi/FEM-BBoption.

2 What is an option and what is it for?

An *option* is a contract which allows to its *holder* at a prescribed time in the future, known as *expiry* or maturity, to buy (in the case of a *call option*) or sell (*put option*) a prescribed *underlying asset* for a prescribed price, called *strike price*. In the nature of the contract is that the holder has the right but no obligation, to exercise the his option; on the other hand, there must be another party in the contract, called the *writer*, who does have the potential obligation to sell (call) or buy (put) the asset at the strike price, even if the trade is not favourable. The option must therefore have some value as it grants a potential gain to the holder with no obligation; on the other hand, the writer must be compensated for the potential liability he has assumes. The theory of *option pricing* is concerned with computing the value options, that is the price a rational investor would be willing to pay in order to have the right granted by an option (and a rational writer would ask to assume the obligation connected to said right).

2.1 The value of an option

In the following the value of an option will be denoted by V. It of course depends on the price of the underlying, denoted by S. The value of the option also takes into account how much time is left before expiry, or *time to maturity* which is denoted by $T - t$, where T is the expiry time and $t < T$ is the present time. This dependence is explicit and is denoted as $V = V(S, t)$. Many other financial parameters influence the value of the option through the value of the underlying asset, but these are not considered explicitly.

The payoff function At maturity, when $t = T$, the holder of an option can check the current price S of the asset and decide weather to exercise the option or not. In the case of a call option, for example, if the current price is higher that the strike price K, the decision

is easy: the holder can buy the asset at the strike price and make a profit selling it on the market at current price. Contrarily if the current price is lower than the strike price the option is worthless. The value of the option at maturity is called *payoff* and is expressed, for a call option as

$$V^{\text{call}}(S,T) = \max\{S-K,0\}\,.$$

Similarly one can find that in the case of a put option the payoff is

$$V^{\text{put}}(S,T) = \max\{K-S,0\}\,.$$

2.2 Uses of options

Options have two primary uses: speculation and hedging. If an investor thinks that the value a given asset is going to raise, he can purchase 100€ worth of said asset; in this way also takes the risk of any fluctuation that the asset value might encounter: if the asset loses 10% of its value, the investor loses 10% of the investment, similarly if the asset value raises. Let us now assume that the investor buys instead call options on the same asset for 10€ with a strike price of 110€: now if the price stays below 110€ he loses 100% of the initial investment, while if the asset value gets to (or above) 110€ the gains again 100% of the initial investment. This phenomenon is called *gearing* and shows how options can be used to expose a portfolio to a greater amount of risk[1]

If on the other hand the investor owns an asset, he can insure against temporary falls of the asset value buying a put option, without the need to liquidate the asset, which might raise again in the future. This is practice is called *hedging* and plays a key role in modern financial engineering.

2.3 Other kinds of options

So far we have discussed only one of the many kind of financial options, that is *European option*, which are the simplest example, and for this reason are called *vanilla options*. *American option* are very similar but the option can be exercised at any time within expiry.

Path-dependent options Other more *exotic* kinds of options are *barrier options* and *Asian options*, whose value is *path-dependent*, in which it does not just depend on the price of the asset at maturity, but on its history from writing to expiry.

The different types of barrier options Barrier options are options whose existence is subject to the crossing of a *barrier value* B by the price of the underlying asset. There are essentially four types of barrier options: *up-and-in*, *up-and-out*, *down-and-in*, *down-and-out*. These all have the property that the right to exercise the option either appears (in

[1] To better understand thinks what would happen if the investor bought 100€ worth of the same options.

the case of *in* barrier) or disappear (*out*) when S reaches some value above (*up*) or below (*down*) the current price. For each barrier we can of course have call or put options, with strike price below or above the barrier.

Basket options A common way to hedge or risk exposure in options portfolios is to write options on various assets or various times: these are called *basket options*. In this case the value of the option at maturity is a prescribed function of the prices of the underlying assets.

3 Basic theory of option pricing: the Black-Scholes model

The most common model for option pricing is due to Fischer Black and Myron Scholes [4], which is based on some substantial assumptions. We will will simply suppose that the asset price follows a geometric Brownian motion and that the market is efficient and frictionless, but a more thorough discussion of all hypotheses to the Black-Scholes model can be found in [12].

3.1 The random nature of market

Τύχη first gained her place in financial theory when Bachelier, in his doctoral thesis [2], modelled unit variations in the prices of assets traded in *Bourse* of Paris as steps of a coin tossing game. Bachelier's work was remarkably original, an anticipation of modern econophysics, but was not much appreciated by his contemporaries and was hugely neglected for decades [9]. The same idea of a *fair game* underlying all finBoth when used for gearing and for hedging, options are ancial trades recursed in many isolated instances throughout the years, and today sturdily sits at the very core of orthodox financial theory [9].

In this view we suppose that asset prices follow a geometric Brownian motion, that is S satisfies the following stochastic differential equation

$$\mathrm{d}S = \underbrace{rS\mathrm{d}t}_{\text{Deterinistic}} + \underbrace{\sigma S\mathrm{d}W}_{\text{Stochastic}} . \tag{1}$$

The first contribution to the variation of the asset price is predictable and expresses the return of money invested in a risk free portfolio: r is the *risk-free interest rate* (the financial matter here gets think, we suggest to see [7] for a thorough discussion). The second contribution comes from all unpredictable fluctuations of the market, such as the response to unexpected news. It is represented by a random sample drawn from a normal distribution of mean zero and variance equal to the time step length $\mathrm{d}t$; σ is the variance, or *volatility* of the price variation per time step.

3.2 From randomness to determinicity

Using some classical machinery of stochastic calculus, namely Itô's lemma and some expansions, we can transform the random contribution in (1) into a deterministic diffusion process, getting to the Black-Scholes equation for option value $V(S,t)$:

$$\frac{\partial V}{\partial t} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} + rS\frac{\partial V}{\partial S} - rV = 0, \tag{BS}$$

for $t \in [0,T]$ and $S \in [0,\infty)$.

As already noted by Bachelier [2], the evolution of options prices is somehow not dissimilar from a heat diffusion process, and eq. (BS) can be transformed into the classical heat equation by means of a change of variables. It is therefore possible, with classical theory of parabolic PDEs, to derive an analytic solution for the Black-Scholes model (BS), which can be found in [12]. The existence of an analytic solution, nonetheless, does not make numerical methods useless, as the evaluation of said solution is, by itself, a non trivial task.

Boundary and final conditions Since the value of the option is known at maturity, final conditions are simply given by the payoff functions discussed in section 2.1. Boundary conditions are instead derived from the *put-call parity*, and can be found in [12], but are not needed in the rest of this paper.

4 Two-asset double-barrier options under the Black-Scholes model

The Black-Scholes model can be generalised to basket options using the multidimensional Itô's lemma. For the value of an option on two underlying assets $S_1, S_2 \in \Omega$ with maturity T the Black-Scholes equation is

$$\frac{\partial V}{\partial t} + \frac{1}{2}\sum_{i,j=1}^{2} \rho_{ij}\sigma_i\sigma_j S_i S_j \frac{\partial^2 S}{\partial S_i\, \partial S_j} + \sum_{j=1}^{2} rS_j \frac{\partial V}{\partial S_j} - rV = 0, \tag{BBS}$$

for

$$\mathbf{S} = (S_1, S_2) \in \Omega, \quad t \in [0,T],$$

where $\rho_{ij}\sigma_i\sigma_j$ are the entries of the covariance matrix [11]: σ_i is the volatility of S_i, ρ_{ij} is the covariance of S_i with S_j (of course $\rho_{ii} = 1$), for $i,j = 1,2$.

We consider a call option on a basket of two assets with two *knock-out* barriers. The barriers are *down-and-out* at B_1 and *up-and-out* at B_2, that is $V(S_1,S_2,t) = 0$ for $\{S_1 + S_2 \leq B_1\} \cup \{S_1 + S_2 \geq B_2\}$. The domain is therefore delimited by the lines $S_1 + S_2 = B_1$ and $S_1 + S_2 = B_2$, and of course by $S_1 = 0$ and $S_2 = 0$, as represented in fig. 1.

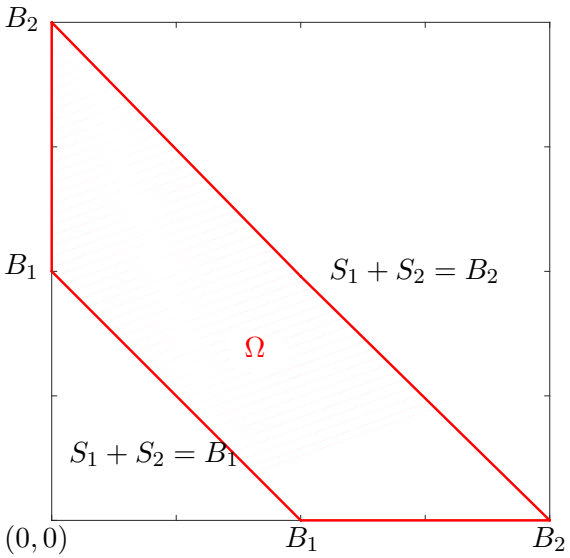


Figure 1 – Domain of the two-assets douele barrier option problem

Boundary and final conditions for the double-barrier option Conditions on two segments of the boundary are imposed by the barriers of the options

$$V(S_1, S_2, t) = 0 \quad \text{on } \{S_1 + S_2 = B_1\} \cup \{S_1 + S_2 = B_2\}.$$

The values for $S_1 = 0$ and $S_2 = 0$ are known by the one-dimensional Black-Scholes equation for an option with two barriers, obtained by simply letting $S_1, S_2 \to 0$ in (BBS). The value of the single asset double-barrier option is known in closed form [6], even if, again, the numerical evaluation is not trivial. We will denote $\bar{V} = V|_{\partial\Omega}$.

The final condition, which is the payoff of the option, depends on the specific contract; we will generally denote the final condition as $V(S_1, S_2, T) = V_T(S_1, S_2)$. A sensible payoff function is $V_T(S_1, S_2) = \max\{S_1 + S_2 - K, 0\}$, where K is the strike price.

4.1 Weak formulation of (BBS)

Consider the following second order partial differential operator

$$\mathcal{L} = \frac{1}{2} \sum_{i,j=1}^{2} \rho_{ij}\sigma_i\sigma_j \frac{\partial^2}{\partial S_i \, \partial S_j} + \sum_{j=1}^{2} r S_j \frac{\partial}{\partial S_j}$$

which can be recognised to be a part of the Black-Scholes operator in (BBS). We write $\mathcal{L}$ in divergence form as

$$\mathcal{L} = \sum_{i=1}^{2} \frac{\partial}{\partial S_i} \left[\frac{1}{2} \sum_{j=1}^{2} \rho_{ij}\sigma_i\sigma_j S_i S_j \frac{\partial}{\partial S_j} \right] - \sum_{i=j=1}^{2} \underbrace{\rho_{jj}}_{1} \sigma_j^2 S_j \frac{\partial}{\partial S_j} - \frac{1}{2} \sum_{\substack{i,j=1 \\ i\neq j}}^{2} \rho_{ij}\sigma_i\sigma_j S_j \frac{\partial}{\partial S_j} + \sum_{j=1}^{2} r S_j \frac{\partial}{\partial S_j};$$

and with a more compact matrix notation

$$\mathcal{L} = \nabla \cdot \mathcal{D}\nabla + b \cdot \nabla$$

where

$$[\mathcal{D}]_{i,j} = \frac{1}{2}\rho_{ij}\sigma_i\sigma_j S_i S_j, \quad [b]_j = S_j \left(r - \sigma_j^2 - \frac{1}{2} \sum_{\substack{i=1 \\ i\neq j}}^{2} \rho_{ij}\sigma_i\sigma_j \right), \quad i,j = 1,2.$$

Therefore eq. (BBS) is written in divergence form as

$$\frac{\partial V}{\partial t} + \nabla \cdot \mathcal{D}\nabla V + b \cdot \nabla V - rV = 0 \tag{2}$$

Let us now consider a suitable test function ψ and integrate (2) against it over the domain Ω:

$$\int_\Omega \frac{\partial V}{\partial t}\psi \mathrm{d}\mathbf{S} + \int_\Omega \nabla \cdot \mathcal{D}\nabla V \psi \mathrm{d}\mathbf{S} + \int_\Omega b \cdot \nabla V \psi \mathrm{d}\mathbf{S} - \int_\Omega rV\psi \mathrm{d}\mathbf{S} = 0$$

taking ψ compactly supported in Ω, the divergence theorem gives the weak form of (2)

$$\int_\Omega \frac{\partial V}{\partial t}\psi \mathrm{d}\mathbf{S} - \int_\Omega \mathcal{D}\nabla V \cdot \nabla\psi \mathrm{d}\mathbf{S} + \int_\Omega b \cdot \nabla V \psi \mathrm{d}\mathbf{S} - \int_\Omega rV\psi \mathrm{d}\mathbf{S} = 0. \tag{3}$$

In order for the above equations to be meaningful we take $V(S_1, S_1, \cdot)$ and $\psi(S_1, S_2)$ in the space

$$H(\Omega) = \left\{ v : v \in L^2(\Omega),\ S_i \frac{\partial v}{\partial S_i} \in L^2(\Omega),\ i = 1,2 \right\}. \tag{4}$$

which is a Hilbert space (as suggested in [1]) with the following induced norm

$$\|u\|_{H(\Omega)} = \left(\|u\|^2_{L^2(\Omega)} + \sum_{i=1}^{2} \left\| S_i \frac{\partial u}{\partial S_i} \right\|^2_{L^2(\Omega)} \right)^{\frac{1}{2}}. \tag{5}$$

In particular the test function ψ is taken in the closure $H_0(\Omega)$ of the space of smooth functions with compact support in Ω with respect to the topology of $H(\Omega)$.

Defining now the bilinear form on $H(\Omega)$

$$a(u,v) = \int_\Omega \mathcal{D}\nabla u \cdot \nabla v \mathrm{d}\mathbf{S} - \int_\Omega b \cdot \nabla uv \mathrm{d}\mathbf{S} + \int_\Omega ruv \mathrm{d}\mathbf{S} \tag{6}$$

changing sings in (3), the weak form of the Black-Scholes pricing problem (BBS) is

$$\text{Find } V \in H(\Omega) \quad : \quad -\int_\Omega \frac{\partial V}{\partial t}\psi \mathrm{d}\mathbf{S} + a(V,\psi) = 0 \quad \forall \psi \in H_0(\Omega) \tag{WBBS}$$

such that

$$\begin{aligned} V(S_1,S_2,T) &= V_T(S_1,S_2) \quad (S_1,S_2) \in \Omega \\ V(S_1,S_2,t) &= \bar{V}(S_1,S_2,t) \quad (S_1,S_2) \in \partial\Omega, t \in [0,T]. \end{aligned}$$

Proposition 1. *The weak problem* (WBBS) *admits a unique solution.*

Proof. We verify continuity and weak coercivity of the bilinear form $a(\cdot,\cdot)$. Existence and uniqueness follow from the Lions-Magenes theorem, being H a Hilbert space [1].

Let $u, v \in H(\Omega)$

$$\begin{aligned} |a(u,v)| &\leq \left|\int_\Omega \mathcal{D}\nabla u \cdot \nabla v \mathrm{d}\mathbf{S}\right| + \left|\int_\Omega b \cdot \nabla uv \mathrm{d}\mathbf{S}\right| + \left|\int_\Omega ruv \mathrm{d}\mathbf{S}\right| \\ &\leq \int_\Omega \sum_{j=1}^{2}\sum_{i=1}^{2} |\rho_{ij}\sigma_i\sigma_j| \left|S_j\frac{\partial u}{\partial S_j}\right| \left|S_i\frac{\partial v}{\partial S_i}\right| \mathrm{d}\mathbf{S} \\ &\qquad + \int_\Omega \sum_{j=1}^{2} \left| r - \sigma_j^2 - \frac{1}{2}\sum_{\substack{i=1 \\ i\neq j}} \rho_{ij}\sigma_i\sigma_j \right| \left|S_j\frac{\partial u}{\partial S_j}\right| |v| \mathrm{d}\mathbf{S} \\ &\qquad + \int_\Omega r|u||v| \mathrm{d}\mathbf{S} \\ &\leq r\|u\|_{L^2(\Omega)}\|v\|_{L^2(\Omega)} + \lambda \sum_{i,j=1}^{2} \left\|S_j\frac{\partial u}{\partial S_j}\right\|_{L^2(\Omega)} \left\|S_i\frac{\partial v}{\partial S_i}\right\|_{L^2(\Omega)} \\ &\leq M\|u\|_{H(\Omega)}\|v\|_{H(\Omega)} \end{aligned}$$

which proves continuity.

Let us now consider

$$\int_\Omega \mathcal{D}\nabla u \cdot \nabla u \mathrm{d}\mathbf{S} + \int_\Omega ru^2 \mathrm{d}\mathbf{S} = \sum_{i,j=1}^{2} \int_\Omega \rho_{ij}\sigma_i\sigma_j S_j\frac{\partial u}{\partial S_j} S_i\frac{\partial u}{\partial S_i} \mathrm{d}\mathbf{S} + r\|u\|^2_{L^2(\Omega)},$$

since the covariance matrix $\rho_{ij}\sigma_i\sigma_j$ is positive semi-definite there exist a constant $\lambda \geq 0$ such that

$$\int_\Omega \mathcal{D}\nabla u \cdot \nabla u \mathrm{d}\mathbf{S} + \int_\Omega r u^2 \mathrm{d}\mathbf{S} \geq \lambda \sum_{i,j=1}^{2} \left\| S_j \frac{\partial u}{\partial S_j} \right\|_{L^2(\Omega)} \left\| S_i \frac{\partial u}{\partial S_i} \right\|_{L^2(\Omega)} + r \|u\|^2_{L^2(\Omega)} \geq \alpha \|u\|^2_{H(\Omega)}$$

On the other hand we have that using Green's formulas

$$\begin{aligned} \int_\Omega b \cdot \nabla u u \mathrm{d}\mathbf{S} &= \frac{1}{2} \int_\Omega b \cdot \nabla \left(u^2\right) \mathrm{d}\mathbf{S} \\ &= \frac{1}{2} \left(\int_{\partial\Omega} u^2 b \cdot \mathbf{n} \mathrm{d}\mathbf{S} - \int_\Omega u^2 \nabla \cdot b \mathrm{d}\mathbf{S} \right), \end{aligned}$$

where $\mathbf{n}$ is the unit vector normal to the boundary of Ω; therefore

$$\left| \int_\Omega b \cdot \nabla u u \mathrm{d}\mathbf{S} \right| \leq \frac{1}{2} \int_{\partial\Omega} u^2 \|b\| \underbrace{\|\mathbf{n}\|}_{1} \mathrm{d}\mathbf{S} + \frac{1}{2} \int_\Omega u^2 \|\nabla \cdot b\| \mathrm{d}\mathbf{S}$$

and since b is smooth enough on Ω and its boundary, by trace theorem (see [5]), there exist a positive constant γ such that

$$\left| \int_\Omega b \cdot \nabla u u \mathrm{d}\mathbf{S} \right| \leq \gamma \|u\|^2_{L^2(\Omega)} \leq \gamma \|u\|_{H(\Omega)}$$

and therefore

$$a(u,u) \geq \alpha \|u\|^2_{H(\Omega)} - \gamma \|u\|^2_{H(\Omega)},$$

which proves weak coercivity of $a(\cdot,\cdot)$. □

4.2 Galerkin spatial approximation

Let us consider a family W_h of subspaces of $H(\Omega)$, depending on a spatial discretization parameter h, such that

$$\forall u \in H, \quad \inf_{w_h \in W_h} \|v - w_h\|_{H(\Omega)} \to 0.$$

We can approximate the weak problem (WBBS) in W_h and get the following *Galerkin problem*

$$\text{Find } V_h \in W_h \quad : \quad -\int_\Omega \frac{\partial V}{\partial t} \psi \mathrm{d}\mathbf{S} + a(V,\psi) = 0 \quad \forall \psi \in W_h \cap H_0(\Omega). \tag{GBBS}$$

Since continuity and weak coercivity of the bilinear form have been proved, in theory, the approximation $V_h \in W_h$ converges to the exact solution $V \in H(\Omega)$ of (WBBS).

Let $\{\phi_j\}_{j\in\mathcal{I}}$ be a basis for W_h, with this notation the approximated solution is expressed as

$$V_h(S_1, S_2, t) = \sum_{j\in\mathcal{I}} \omega_j(t)\phi_j(S_1, S_2) \tag{7}$$

where the weights $\omega_j(t)$ are to be determined. The index set $\mathcal{I}$ can be partitioned into two subsets $\mathcal{I} = \mathsf{Dir} \cup \mathsf{Ind}$, where function indexed by Dir are those whose support touches the boundary, and therefore their weight in V_h can be determined by the Dirichlet boundary condition. Functions indexed by Ind are instead compactly supported in Ω, so they span $W_h \cap H(\Omega)$, and their weights are to be determined.

The Galerkin problem (GBBS) is then written with respect to this basis as

$$-\sum_{j\in\mathsf{Ind}\cup\mathsf{Dir}} \dot{\omega}_j(t) \int_\Omega \phi_j\phi_i \mathrm{d}\mathbf{S} + \sum_{j\in\mathsf{Ind}\cup\mathsf{Dir}} \omega_j(t) a\left(\phi_j, \phi_i\right) = 0, \quad \forall i \in \mathsf{Ind}. \tag{8}$$

The matrices

$$[M]_{ij} = \int_\Omega \phi_j\phi_i \mathrm{d}\mathbf{S}, \quad [A]_{ij} = a\left(\phi_j, \phi_i\right), \qquad i \in \mathsf{Ind}, j \in \mathsf{Ind}\cup\mathsf{Dir} \tag{9}$$

are respectively called *mass* and *stiffness* matrix. Equation (8) is then written with compact matrix notation

$$-M\dot{\boldsymbol{\omega}}(t) + A\boldsymbol{\omega}(t) = 0, \tag{10}$$

where $[\boldsymbol{\omega}(t)]_j = \omega_j(t)$ for $j \in \mathsf{Ind}\cup\mathsf{Dir}$.

4.3 Time discretization by θ-method

In order to get from problem (WBBS) to (10) a spatial discretization was performed, as the approximation spaces W_h are immersed in $H(\Omega)$ which only takes into account spatial variables S_1, S_2. Since all the formulations of the discrete problem still contain a time derivative, we perform a time discretization using a θ-method.

Let Δt be a fixed time step, which is the time discretization parameter, we can then approximate the time derivative by an incremental ratio

$$\dot{\boldsymbol{\omega}}(t) \approx \frac{\boldsymbol{\omega}(t+\Delta t) - \boldsymbol{\omega}(t)}{\Delta t}$$

and we write (10) as

$$-M\left[\frac{\boldsymbol{\omega}(t+\Delta t) - \boldsymbol{\omega}(t)}{\Delta t}\right] + A\left[\theta\boldsymbol{\omega}(t+\Delta t) + (1-\theta)\boldsymbol{\omega}(t)\right] = 0,$$

from which the following discrete evolution equation is derived

$$\left(\frac{M}{\Delta t} + (1-\theta)A\right)\boldsymbol{\omega}(t) = \left(\frac{M}{\Delta t} - \theta A\right)\boldsymbol{\omega}(t+\Delta t), \tag{11}$$

which expresses the value of the option at time t given the value at a successive time step. Since the final condition $V(S_1, S_2, T) = V_T(S_1, S_2)$ is known, the above equation can be used as an iterative procedure to actualize the final value of the option at $t = T$ to the present time $t = 0$ through a sequence of time steps. It is worth noticing that since the problem is in backward form, the method is unstable for $\theta > \frac{1}{2}$.

Since weights indexed by Dir, denoted by the sub-vector $\boldsymbol{\omega}_{\mathsf{Dir}}$, are known from the boundary conditions, we should find a system for only the independent weights $\boldsymbol{\omega}_{\mathsf{Ind}}$. Let us expand the matrix product in (11)

$$\sum_{j \in \mathsf{Ind} \cup \mathsf{Dir}} \left(\frac{M_{ij}}{\Delta t} + (1-\theta) A_{ij} \right) \omega_j(t) = \sum_{j \in \mathsf{Ind} \cup \mathsf{Dir}} \left(\frac{M_{ij}}{\Delta t} + \theta A_{ij} \right) \omega_j(t + \Delta t)$$

and splitting the sum over the two index sets we get

$$\begin{aligned} \sum_{j \in \mathsf{Ind}} \left(\frac{M_{ij}}{\Delta t} + (1-\theta) A_{ij} \right) \omega_j(t) = & \sum_{j \in \mathsf{Ind} \cup \mathsf{Dir}} \left(\frac{M_{ij}}{\Delta t} + \theta A_{ij} \right) \omega_j(t + \Delta t) \\ & - \sum_{j \in \mathsf{Dir}} \left(\frac{M_{ij}}{\Delta t} + (1-\theta) A_{ij} \right) \omega_j(t). \end{aligned}$$

In order to write in matrix form the previous equation we denote

$$M_{\mathsf{Ind}} = [M_{ij}]_{\substack{i \in \mathsf{Ind} \\ j \in \mathsf{Ind}}}, \quad A_{\mathsf{Ind}} = [A_{ij}]_{\substack{i \in \mathsf{Ind} \\ j \in \mathsf{Ind}}}$$

the square sub-matrices of M and A respectively formed by only the columns indexed in Ind and

$$M_{\mathsf{Dir}} = [M_{ij}]_{\substack{i \in \mathsf{Ind} \\ j \in \mathsf{Dir}}}, \quad A_{\mathsf{Dir}} = [A_{ij}]_{\substack{i \in \mathsf{Ind} \\ j \in \mathsf{Dir}}}.$$

With this notation we finally get to the discrete evolution equation

$$\begin{aligned} \left(\frac{M_{\mathsf{Ind}}}{\Delta t} + (1-\theta) A_{\mathsf{Ind}} \right) \boldsymbol{\omega}_{\mathsf{Ind}}(t) = & \left(\frac{M}{\Delta t} - \theta A \right) \boldsymbol{\omega}(t + \Delta t) \\ & - \left(\frac{M_{\mathsf{Ind}}}{\Delta t} + (1-\theta) A_{\mathsf{Ind}} \right) \boldsymbol{\omega}_{\mathsf{Dir}}(t) \end{aligned} \tag{12}$$

which allows to only compute independent weights.

5 Finite element approximation of the weak Black-Scholes problem

In the finite element method the geometry is divided into simple polygons, in our case triangles, and the approximation space consists of piecewise polynomial functions. Let $\mathcal{T}_h$

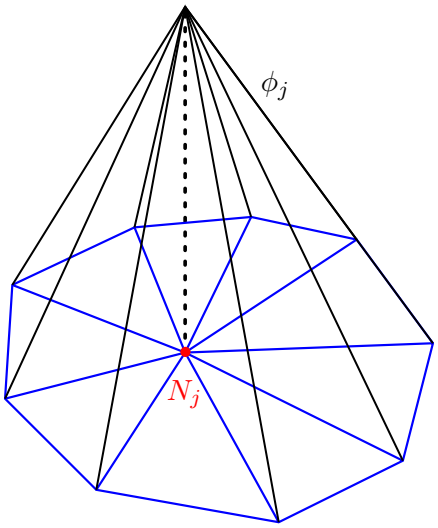


Figure 2 – Representation of a FEM linear basis function.

be a *triangulation* of the domain Ω, that is a set of non degenerate triangles such that

$$\bar{\Omega} = \bigcup_{K \in \mathcal{T}_h} K.$$

The triangulation is said to be *admissible* if the interiors are disjoint and the intersections of each two triangles only are vertices or edges. The mesh fineness is described by the discretization parameter h which can be fixed to be $h = \sup_{K \in \mathcal{T}_h} \mathsf{diam} K$.

Given an admissible triangulation $\mathcal{T}_h$, the approximation in the FEM framework is built in a piecewise polynomial space

$$\mathrm{X}_h^r = \left\{ v_h \in \mathcal{C}^0\left(\bar{\Omega}\right) : v_h|_K \in \mathbb{P}^r \ \forall K \in \mathcal{T}_h \right\} \tag{13}$$

which is a subspace of H. In this paper only linear approximations are used, that is $r = 1$.

As a basis of X_h^1 we chose a set of locally supported functions $\{\phi_i\}_{i \in \mathcal{I}}$, in order to have less non-zero elements in the stiffness matrix computation; the basis is also chosen to be lagrangian, that is $\phi_j(\mathbf{p}_j) = \delta_{ij}$ for all nodes $\mathbf{p}_j$ of the triangulation, as in fig. 2. With this choice of the basis, the weights in the linear combination (7) are exactly the values of V in the nodes of the triangulation:

$$\omega_j(t) = V(\mathbf{p}_j, t).$$

5.1 Geometry discretization

The first step for the FEM approximation procedure is the discretization of the geometry of the domain with a triangulation.

5.1.1 Geometry description

We will assume that the geometry is a convex polygon, as such is our domain in fig. 1, and variations that we could be interested in. In our implementation the polygon is described by its vertices written in Cartesian coordinates and two column matrix which contains a list of the edges of the polygon. Furthermore in order to keep track of boundary conditions, each edge is assigned a unique identifier. For a concrete view the reader can take a look at the data structure implemented in the `poly_geo` class.

5.1.2 Mesh generation and refinement

As a mesh we used a standard non-constrained Delaunay triangulation, built with the algorithm implemented in the `Qhull` library [3], which is included in MATLAB . The triangulation consists of a list of *nodes*, the vertices of the triangles, and a *connectivity matrix* which lists the indices of the vertices of each triangles in positive order following the right hand rule. Another important datum that we need to keep track of is what edges of the triangles lie on the boundary of the domain. In our any edge of the triangulation that lie on a portion of the boundary inherit the same identifier.

A first, rough, triangulation is obtained simply computing the Delaunay triangulation of the vertices of the geometry. Additional nodes can be added manually to improve the behaviour of the triangles (peep ahead fig. 5 to see why). To refine the mesh a very simple splitting technique is used: a node is added at the midpoint of every edge, and a new Delaunay triangulation is computed. Boundary identifiers for every new node are kept track of only using geometrical properties. Edges that are on the boundary of the unrefined mesh are substituted in the list of boundary edges with the two segments in which the node insertion splits the edge and both segments inherit the same boundary identifier of the unrefined edge (see the implementation in the `midsplitref` method of the `@tri_mesh` class).

5.2 Lagrangian basis for the approximations space

Now that the triangulation $\mathcal{T}_h$ has been built, the approximation space X_h^1 in (13) is explicitly defined. The basis again is simply given by the *tent functions* represented in fig. 2. As again in fig. 2 the basis function ϕ_j, which only insists on the node $\mathbf{p}_j$, can be decomposed into a different linear function on every triangle that shares the same node $\mathbf{p}_j$.

5.2.1 Local nodal basis

The vertices of any triangle $K \in \mathcal{T}_h$ are denoted by $\mathbf{p}_1^K, \mathbf{p}_2^K, \mathbf{p}_3^K$. The correspondence between local indices $1, 2, 3$ and the global indices of the three nodes in the triangulation is found in the connectivity matrix: in the row corresponding to K, the first entry is the global index of the first vertex, and so on.

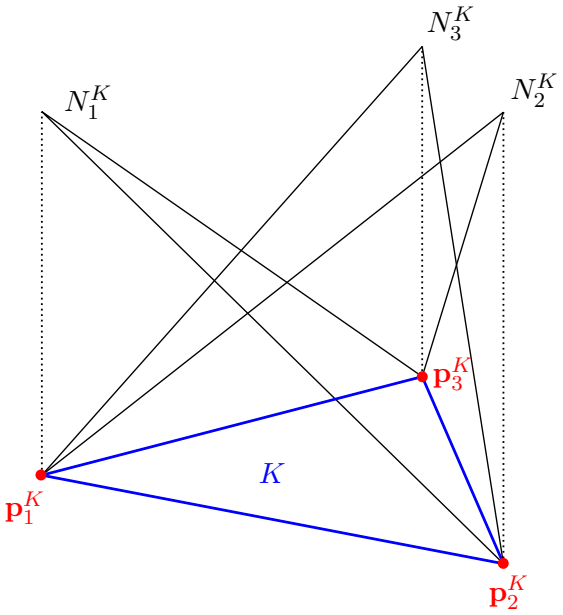


Figure 3 – Schematic view of the local nodal basis in K.

We define for each triangle K three local nodal basis functions N_1^K, N_2^K, N_3^K such that $N_i^K(\mathbf{p_j^K}) = \delta_{ij}$ for all $i, j = 1, 2, 3$, as in fig. 3.

As already mentioned, for any basis function of X_h^1, ϕ_j, where the global index j corresponds to the vertex $\mathbf{p}_i^K$ of K, we have that

$$\phi_j|_K = N_i^K.$$

5.2.2 Reference element

In the following steps we need to evaluate the local nodal basis functions, to make thing cleaner we define the *reference element* $\hat{K}$, that is the unitary triangle of $\mathbb{R}^2$ represented to the left in fig. 4, where $\hat{\mathbf{p}}_1 = (0,0), \hat{\mathbf{p}}_2 = (1,0), \hat{\mathbf{p}}_3 = (0,1)$.

Each triangle of $\mathcal{T}_h$ can be obtained from $\hat{K}$ by means of a simple linear affine transformation

$$T_K(x,y) = B_K \begin{bmatrix} x \\ y \end{bmatrix} + \mathbf{p}_1^K, \tag{14}$$

where the matrix which deforms $\hat{K}$ into K is written by columns as

$$B_K = \left[\mathbf{p}_2^K - \mathbf{p}_1^K, \mathbf{p}_3^K - \mathbf{p}_1^K\right].$$

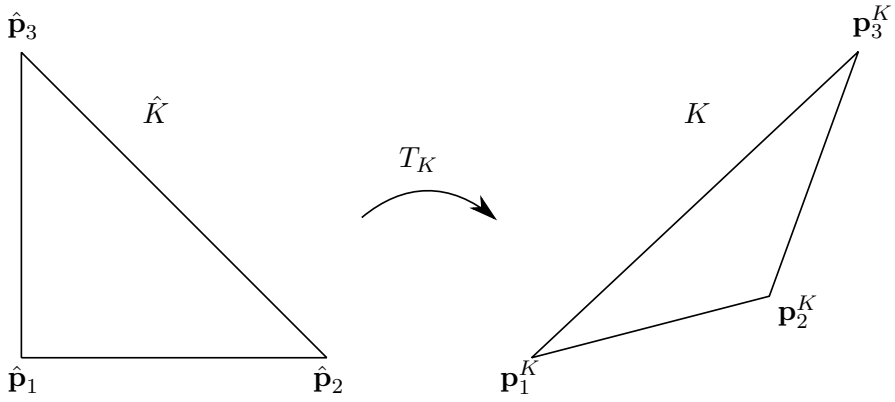


Figure 4 – Correspondence between the reference element $\hat{K}$ and a triangle $K \in \mathcal{T}_h$.

5.2.3 Reference nodal basis

Of course a set of nodal Lagrangian basis functions $\hat{N}_1, \hat{N}_2, \hat{N}_3$ can be defined on the reference element as well:

$$\begin{aligned} &\hat{N}_1 = 1 - x - y, \\ &\hat{N}_2 = x, \qquad\qquad \text{with } (x, y) \in \hat{K}. \\ &\hat{N}_3 = y, \end{aligned} \tag{15}$$

Of course the nodal functions on $K \in \mathcal{T}_h$ can be recovered from these as

$$N_\alpha^K(s_1, s_2) = \hat{N}_\alpha \circ T_K^{-1}(s_1, s_2), \quad \alpha = 1, 2, 3, \tag{16}$$

where

$$T_K^{-1}(s_1, s_2) = B_K - 1 \left(\begin{bmatrix} s_1 \\ s_2 \end{bmatrix} - \mathbf{p}_1^K \right), \quad \begin{bmatrix} s_1 \\ s_2 \end{bmatrix} \in K.$$

Similarly the gradient of all nodal functions on K can be computed from the gradient of the reference nodal functions as

$$\nabla N_\alpha^K(s_1, s_2) = \left[B_K^{-1} \right]^T \nabla \hat{N}_\alpha \left(F_K^{-1}(s_1, s_2) \right), \quad i = 1, 2, 3. \tag{17}$$

On the other hand we notice that

$$\nabla \hat{N}_1 = \begin{bmatrix} -1 \\ -1 \end{bmatrix}, \quad \nabla \hat{N}_2 = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \quad \nabla \hat{N}_2 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}.$$

We have therefore found a very simple way to implement the evaluation of the local nodal basis functions and their gradients for any triangle $K \in \mathcal{T}_h$.

5.3 Mass and stiffness matrices

5.3.1 Mass matrix assembly

The expression for the mass matrix entries in (9) can be decomposed by linearity:

$$[M]_{ij} = \int_\Omega \phi_j \phi_i \mathrm{d}\mathbf{S} = \sum_{K \in \mathcal{T}_h} \int_K \phi_j \phi_i \mathrm{d}\mathbf{S}. \tag{18}$$

On K the integral is non-zero only if both i and j are global indices for one of the vertices of K, which gives nine combination of non-zero entries in the mass matrix.

Local mass matrix Each basis function ϕ_i becomes one of the three local basis functions when restricted to a triangle

$$[M_K]_{\alpha\beta} = \int_K N_\beta^K N_\alpha^K \mathrm{d}\mathbf{S}, \quad \alpha, \beta = 1, 2, 3,$$

which gives a 3×3 matrix called *local mass matrix*. We can easily compute these integrals using the reference element:

$$\int_K N_\beta^K N_\alpha^K \mathrm{d}\mathbf{S} = \int_{\hat{K}} \hat{N}_\beta \hat{N}_\alpha \left|\det B_k\right| \mathrm{d}\mathbf{x} = \det B_k \int_{\hat{K}} \hat{N}_\beta \hat{N}_\alpha \mathrm{d}\mathbf{x},$$

which is easily computed by hand:

$$\int_{\hat{K}} \hat{N}_j \hat{N}_i \mathrm{d}\mathbf{x} = \begin{cases} \frac{1}{12} & \text{if } i = j \\ \frac{1}{24} & \text{if } i \neq j. \end{cases}$$

Finally, after all local mass matrices are computed, eq. (18) allows to construct the global mass matrix, using the aforementioned correspondence between local and global indices. A refined implementation of this procedure can be found in the function `assembly/Mass.m` of the code library.

5.3.2 Stiffness matrix assembly

We now consider eqs. (6) and (9) and decompose the integral over Ω in the sum of local contributions

$$[A]_{ij} = \sum_{K \in \mathcal{T}_h} \int_K \mathcal{D} \nabla \phi_j \cdot \nabla \phi_i \mathrm{d}\mathbf{S} - \int_K b \cdot \nabla \phi_j \phi_i \mathrm{d}\mathbf{S} + \int_K r \phi_j \phi_i \mathrm{d}\mathbf{S} \tag{19}$$

As for the mass matrix, the computation is carried out locally, then the global mass matrix is assembled using the equation above.

Local stiffness matrix On each triangle the previous, using local indices, we compute the *local stiffness matrix* with entries

$$[A_K]_{\alpha\beta} = \int_K \mathcal{D}\nabla N^K_\beta \cdot \nabla N^K_\alpha \mathrm{d}\mathbf{S} - \int_K b \cdot \nabla N^K_\beta N^K_\alpha \mathrm{d}\mathbf{S} + \int_K r N^K_\beta N^K_\alpha \mathrm{d}\mathbf{S}, \tag{20}$$

for $\alpha, \beta = 1, 2, 3$. The last addend is of course just rM_K, where the local mass matrix M_K has already been computed.

Quadrature formula In the first two integrals in (19), the change of variable to the reference element is not useful, since the integrands depend explicitly on S_1, S_2. Fortunately both integrands are of total degree two, so we can use a simple *midpoint rule* to get the exact value of the integral, being the midpoint rule of the second order. The integral on a triangle $K \in \mathcal{T}_h$ is computed with the formula

$$\int_K f(\mathbf{S})\mathrm{d}\mathbf{S} = \frac{\mathrm{Area}(K)}{3} \sum_{\gamma=1}^{3} f(m_\gamma),$$

where the area of the triangle can be computed as $\det B_K/2$ and M_γ are the midpoints of the three edges of K. At this point the computation is pretty straight forward, the implementation is in `assembly/Mass_local.m`.

5.3.3 Dirichlet nodes

In the computation of the mass and stiffness matrices we have so far considered all nodes, but since test functions have been chosen to be compactly supported in Ω, eq. (9) shows that only rows indexed by Ind are to be considered. Similarly eq. (12) requires to separate columns of M and A indexed by Ind and Dir.

The choice of a Lagrangian basis allows us to identify basis functions with nodes, in that for each node one and only one basis function in non-zero no it. Therefore, the index set Dir coincides with the indices of the nodes that lie on the boundary, which are saved in the mesh data. Of course the weights on these nodes are known form the boundary conditions

$$\omega_j(t) = \bar{V}(\mathbf{p}_j, t), \quad j \in \mathsf{Dir}.$$

6 Numerical experiments

In this section we use our finite element library to numerically explore the problem of pricing the two-asset double knock-out European call option introduces in section 4. As a basic we follow [10] setting parameters as in table 1 and and the barriers $B_1 = 1$, $B_2 = 2$. Different parameters will be specified time by time.

Table 1 – Parameters for the basic example

K	T	σ_1	σ_2	ρ	r
1	1	0.25	0.25	0.7	0.05

Boundary value To compute the value on the boundary we used the following formula for a single asset double-knock-out European call adapted from [6]:

$$
\begin{aligned}
V(S,t) = Se^{-rt} \sum_{n=-\infty}^{+\infty} \left\{ \left(\frac{B_2^n}{B_1^n}\right)^{\mu} [N(d_1) - N(d_2)] - \left(\frac{B_1^{n+1}}{SB_2^n}\right)^{\mu} [N(d_3) - N(d_4)] \right\} \\
- Ke^{-rt} \sum_{n-\infty}^{+\infty} \left\{ \left(\frac{B_2^n}{B_1^n}\right)^{\mu-2} \left[N(d_1 - \sigma\sqrt{t}) - N(d_2 - \sigma\sqrt{t})\right] \right. \\
\left. - \left(\frac{B_1^{n+1}}{SB_2^n}\right)^{\mu-2} \left[N(d_3 - \sigma\sqrt{t}) - N(d_4 - \sigma\sqrt{t})\right] \right\},
\end{aligned}
$$

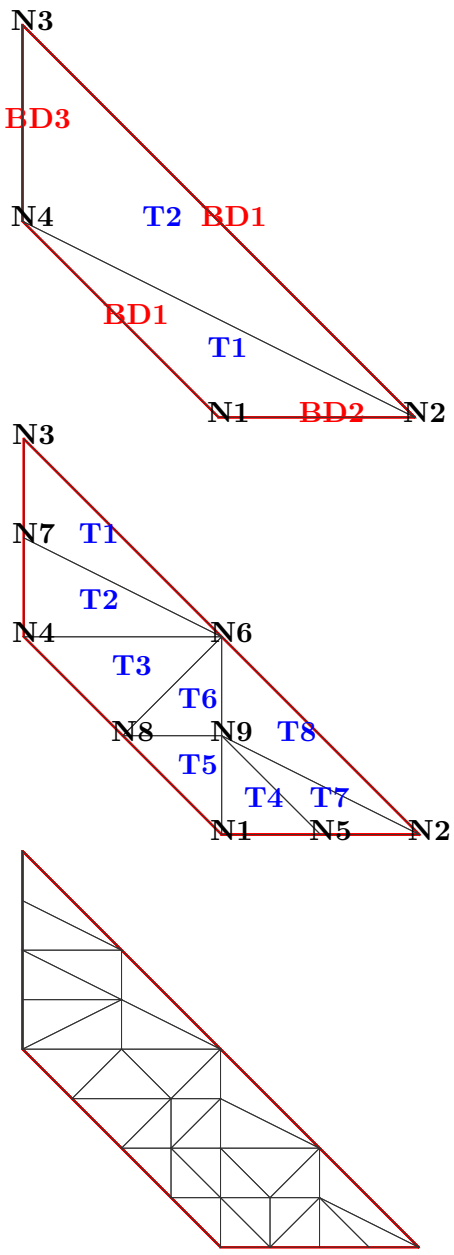
where

$$
\begin{aligned}
\mu &= \frac{2}{\sigma^2} + 1 \\
d_1 &= \frac{\log\left(\frac{SB_2^{2n}}{KB_1^{2n}}\right) + \frac{\sigma^2}{2}t}{\sigma\sqrt{t}} \qquad d_2 = \frac{\log\left(\frac{SB_2^{2n-1}}{B_1^{2n}}\right) + \frac{\sigma^2}{2}t}{\sigma\sqrt{t}} \\
d_3 &= \frac{\log\left(\frac{B_1^{2n+2}}{KSB_2^{2n}}\right) + \frac{\sigma^2}{2}t}{\sigma\sqrt{t}} \qquad d_4 = \frac{\log\left(\frac{B_1^{2n+2}}{SB_2 2n+1}\right) + \frac{\sigma^2}{2}t}{\sigma\sqrt{t}},
\end{aligned}
$$

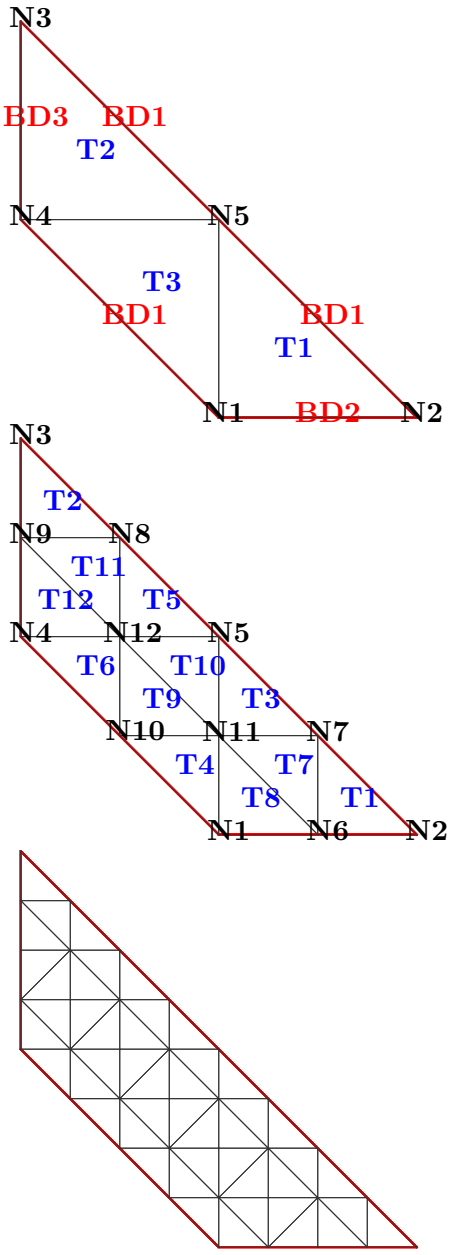
and t is the time to maturity, S is the value of the underlying asset, r the risk-free interest rate and σ the volatility of the asset. In our implementation, found in the function `doubleOUT_call`, the summation has been truncated to the range $-5, \ldots, 5$ as suggested in [6]. No appreciable differences in the output has be observed with summations up to the tenth order. On the other hand it has already been proved numerically in [8] that convergence is fast.

Geometry description and meshing The geometry of the domain is described by the vertices of the polygons and a matrix of links representing the edges. Ad already mentioned, fictitious vertices can be added manually to improve the behaviour of the triangulations. Figure 5 shows two different behaviours in the mesh sequence produces by inserting only one additional node.

The solutions obtained on the two sets of meshes differ significantly, for example in fig. 6. One is not necessarily better than the other, but we generally preferred more predictable structured meshes.

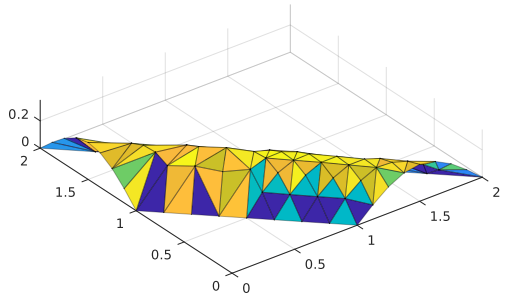


(a) Unstructured meshes obtained by Delaunay triangulation of the vertices of the polygonal domain.

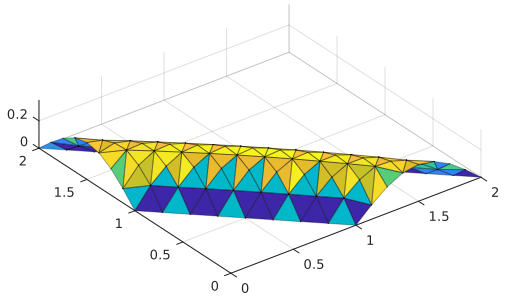


(b) Structured meshes obtained by adding a fictitious vertex at **N5** in the geometry.

Figure 5 – Two refinements of two different meshings of the domain. Nodes are labelled by N indices and triangles are labelled by T. BD indices are the boundary identifiers. The two barriers $S_1 + S_2 = B_1$ and $S_1 + S_2 = B_2$ have the same identifier because they have the same boundary condition.

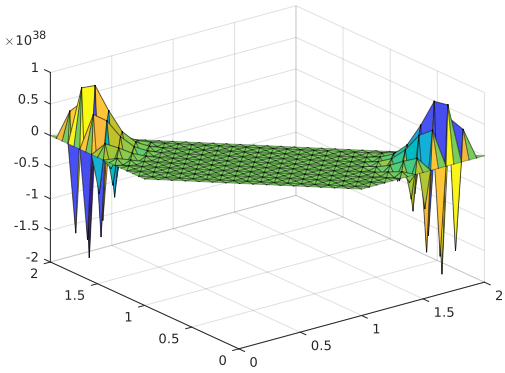


(a) Solution on the unstructured mesh.

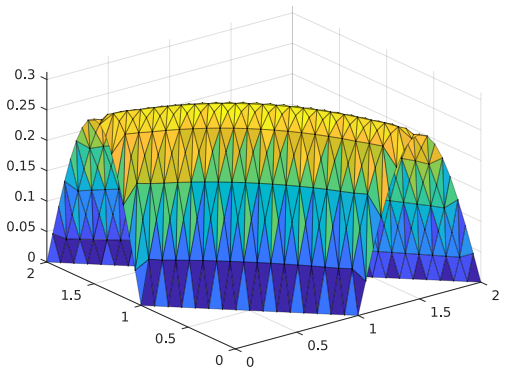


(b) Solution on the structured mesh

Figure 6 – Solutions for the basic example with parameters in table 1 obtained with 100 steps of the backward Euler method ($\theta = 0$) on a mesh refined 3 times.



(a) 100 time steps



(b) 180 time steps

Figure 7 – Stability comparison with $\theta = 1$ on the same mesh with two different time discretizations.

Time discretization When choosing the time discretization we need to consider the problem of stability, we know that θ-methods with $\theta \leq \frac{1}{2}$ are stable, while stability of methods with $\theta > \frac{1}{2}$ depends on the ratio between the time step and the square of the geometric discretization parameter. Therefore refining the mesh might significantly deteriorate the solution, as in fig. 7a. In this case refining the time discretization in necessary to recover stability, as in fig. 7b. Since they do not offer essentially any advantage, we generally avoided methods with $\theta < \frac{1}{2}$.

We do not have a reference solution and it is therefore not easy to state the quality of our approximation, but in table 2 we compared our results to those [10]. Frome the results it seems that the backward Euler method (BEU) is more sensitive to the time discretization, unfortunately the convergence is not monotone. The Crank-Nicolson (CN, $\theta = 0.5$) method instead seem to be less dependent on the time discretization; convergence

Table 2 – Values of $V_h(1.25, 0.25, 0)$ for the basic problems with parameters table 1 approximated with our solver with different discretization parameters. The reference value in [10] is $V(1.25, 0.25, 0) \approx 0.2949$. N_R is the number of mesh refinements, N_t the number of time steps.

N_R	θ	N_t	$V_h(1.25, 0.25, 0)$
3	0	100	0.2906
4	0	100	0.2945
5	0	100	0.2956
4	0	500	0.2938
5	0	500	0.2949
3	0.5	100	0.2897
4	0.5	100	0.2936
5	0.5	100	0.2947
4	0.5	500	0.2936
5	0.5	500	0.2947

with space discretization strangely seems to be slower, bu monotone. Nonetheless the significance of the reference value is not known, so these results are to be taken as they are.

The biggest computational cost is the evaluation of boundary conditions, and since they need to be computed at every time step, we will prefer CN, which has a better convergence with fewer time steps. A reference plot for this example is fig. 8.

6.1 Moving the strike price

In the previous example the down barrier was doing nothing, as it coincided with the strike price, and therefore the value of the option would have been zero anyway. If the strike price is lower the the down barrier the setting is the same. Similarly if the strike price is equal or grater than the up barrier, the value of the option is constantly zero as, whenever it has any financial value, the barrier deactivates it. The values for the strike price where both barriers do something are therefore $B_1 < K < B_2$. Changing the strike price immediately alters the payoff function

$$V_T(S_1, S_2) = \max\{S_1 + S_2 - K, 0\}$$

which is plotted in fig. 9 for some values of K.

Actualizing the final values in fig. 9 with the Crank-Nicolson iteration we regularise the payoff function (fig. 10), the current values are plotted in fig. 7.

Of course we can see that getting further away from the final condition the solution becomes smoother and the option acquires some value even where the payoff is zero. In

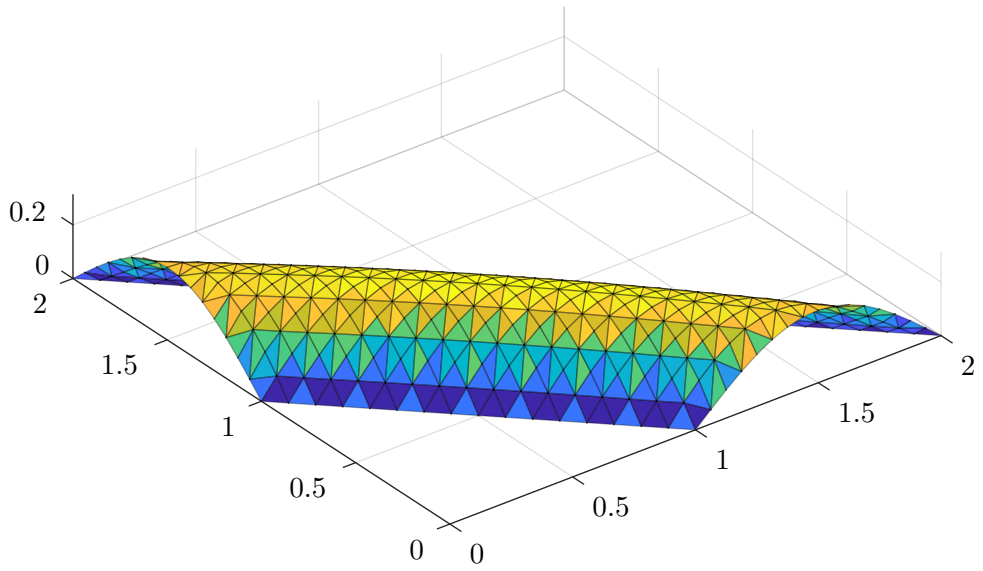
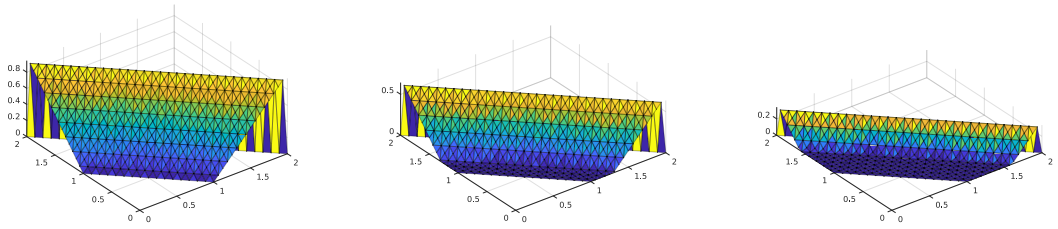
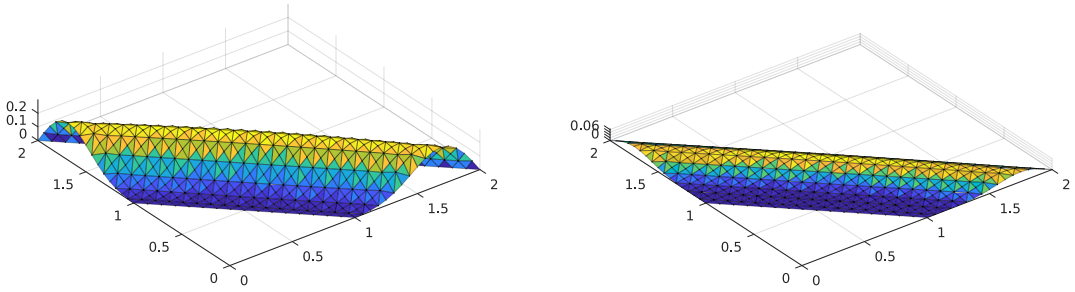


Figure 8 – Solution to the problem with parameters table 1, solved on a mesh refined 4 times with 100 CN steps.



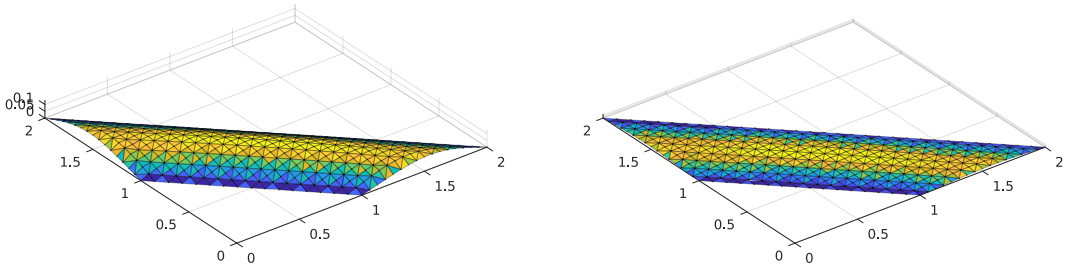
(a) $K = 1$ (b) $K = 1.3$ (c) $K = 1.6$

Figure 9 – Plots of the payoff functions for three values of K.



(a) $K = 1.3$ (b) $K = 1.6$

Figure 10 – Intermediate solution at $t = 0.75$ or different values of K.



(a) $K = 1.3$ (b) $K = 1.6$

Figure 11 – Current value ($t = 0$) for different values of K.

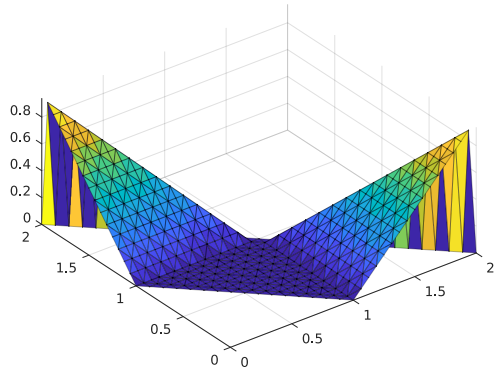


Figure 12 – Payoff (21)

fact with more time from expiry, the probability that the prices of the underlying evolve positively become greater. Nonetheless when the strike price moves towards the up barrier, the overall value of the option decreases, because measure of the support of the payoff is smaller, and therefore the probability of ending the process in that set decreases.

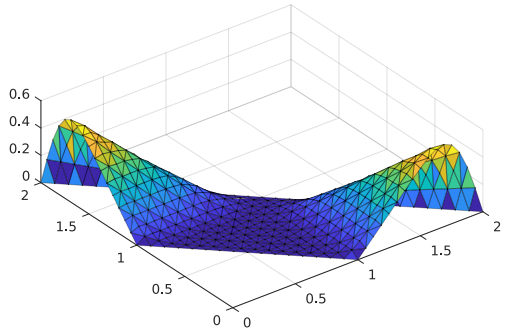
6.2 changing the payoff

The previous example the payoff function was altered by moving the strike price, but in some contracts the payoff can also be a totally different function of S_1, S_2, as long as it links with the boundary conditions.

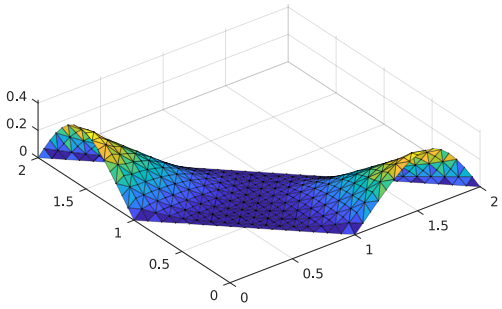
Buying the highest priced underlying Let un consider the following final condition, plotted in

$$V_T(S_1, S_2) = \max\{\max\{S_1, S_2\} - K, 0\}, \tag{21}$$

that is, the option allows the holder to buy the asset that is worth the most. Again the effect of actualizing the final value tends to regularize the final condition (fig. 13), and this

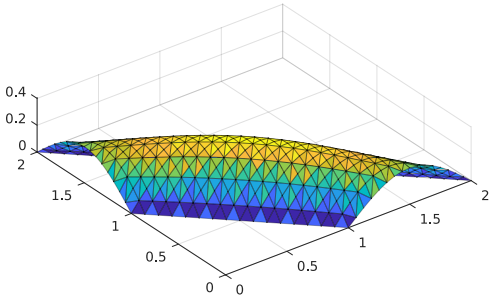


(a) $t = 0.75$

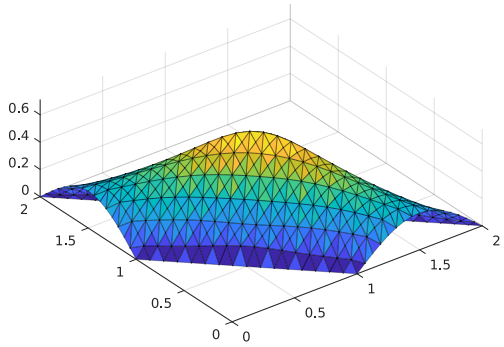


(b) $t = 0.25$

Figure 13 – Intermediate approximations with parameters table 1 with final value (21) obtained with 100 CN steps.



(a) Solution with decorrelated assets.



(b) Solution with anticorrelated assets.

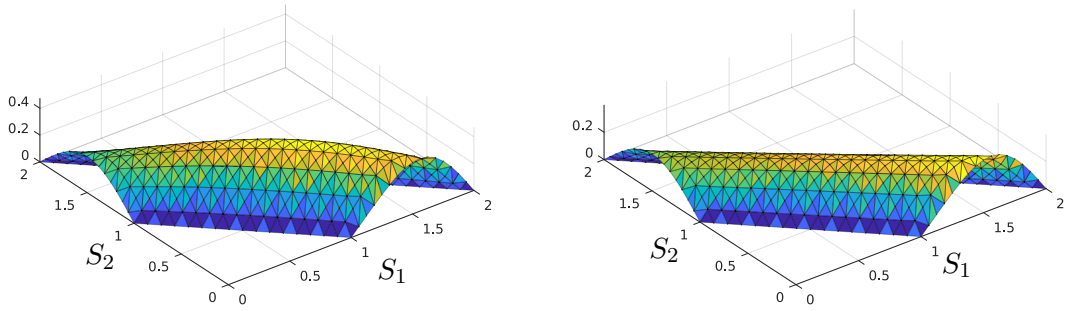
Figure 14 – Effects of decorrelation and anticorrelation.

is expected as the Black-Scholes model is somehow a backward heat equation. ¡

6.3 Different statistical properties of the underlying assets prices

Our previous tests show that the maximum value of the option tend to be well diffused along a line parallel to the barriers. This phenomenon is due to the high correlation between the underlying assets. If we decorrelate the assets setting $\rho = 0$, the maximum tends instead to be centred in graph as in see fig. 14a. If the assets are anticorrelated the mass is even more concentrated, fig. 14b,

Keeping the assets decorrelated, if volatilities are different, we can see in fig. 15a that the *hump* moves and the value of the option is higher for higher prices of the less volatile asset. The value of one-asset double barrier option on the $S_2 = 0$ is higher with less volatility, which marks a difference with plain vanilla options, which are more valuable if



(a) Decorrelated assets: $\rho = 0$.

(b) Correlated assets: $\rho = 1$.

Figure 15 – Option value when assets have different volatilities $\sigma_1 = 0.18,\ \sigma_2 = 0.25$.

the underlying asset is more volatile. The financial reasoning is that with higher volatility there is an higher probability that the asset price increases (since exercising the option is not compulsory, a decrease in the asset price is not worrisome); if we add barriers, higher volatilities increase the probability that the asset price exits the *money* region and cancelling the option.

If the same assets are correlated, the hump disappears, and the mass gets linearly dispersed along the central *spine* of the plot, as fig. 15b.

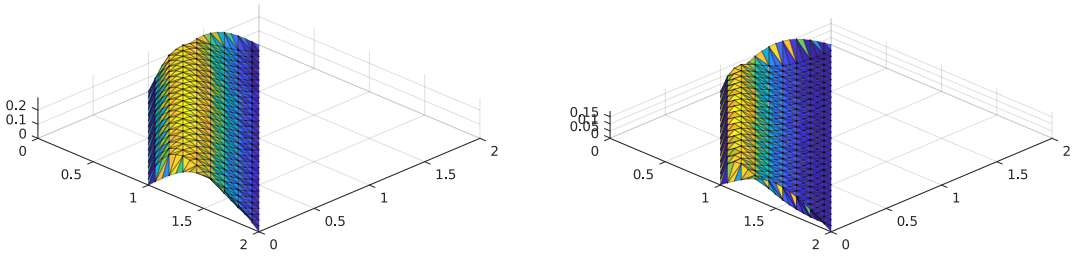
6.4 And what if the bank gave me more?

As discussed in section 2.2, options are used in speculation through gearing, which allows to multiply the amount ov money moved, before an higher risk of loss, and for hedging, writing options of counter-correlated assets. Both of these purposes are defeated if the current risk-free interest rate is very high. We can see in fig. 16 that increasing r moves the maximum values of the options toward the region where the price of the assets is lowest. On the $S_1 = 0$ and $S_2 = 0$ boundaries the option follows the one-dimensional problem. Overall the value of the option is again much lower.

The idea is that if the risk-free return of a given investment is very high, one would be much better off simply investing in there. Only if the price of the option is very low, or if one can buy the underlying assets for very little money it makes sense to go through the option.

7 Conclusions and things that are missing in this paper

In this work we have presented some basic notions of financial options and theory of option pricing. The Black-Scholes model has been discussed and adapted to the problem of pricing an call option depending on two underlying assets with two out barriers. Since this model



(a) $r = 0.2$ (b) $r = 0.5$

Figure 16 – Solutions with high risk-free interest rate.

does not have a closed for solution we developed a FEM library and discussed some details on the implementation.

Our `FEM-BBoption` library showed promising results in the numerical simulations, but we could not find exact data for a comparison. We then used our library to discuss some features of the Black-Scholes model applied to our problem.

Our approach is unfortunately very limited since our codes only allow for convex polygonal domains in 2D. It is worth noting that the FEM would not be suitable for more complex multi-asset financial derivatives anyway, as its complexity grows too fast with the number of geometric dimension.

In our numerical test we briefly discussed the the effects on different statistical properties of the underlying assets on the value of the option, but a proper sensitivity analysis was not performed.

References

[1] Y. Achdou and O. Pironneau. *Computational Methods for Option Pricing*. Society for Industrial and Applied Mathematics, Jan. 2005. DOI: 10.1137/1.9780898717495.

[2] L. Bachelier. "Théorie de la spéculation". fr. PhD thesis. 1900, pp. 21–86. DOI: 10.24033/asens.476.

[3] C. B. Barber, D. P. Dobkin, and H. Huhdanpaa. "The Quickhull Algorithm for Convex Hulls". In: *ACM Trans. Math. Softw.* 22.4 (Dec. 1996), pp. 469–483. ISSN: 0098-3500. DOI: 10.1145/235815.235821.

[4] F. Black and M. Scholes. "The Pricing of Options and Corporate Liabilities". In: *Journal of Political Economy* 81.3 (May 1973), pp. 637–654. DOI: 10.1086/260062.

[5] H. Brezis. *Functional Analysis, Sobolev Spaces and Partial Differential Equations*. Springer New York, 2010. DOI: 10.1007/978-0-387-70914-7.

[6] E. G. Haug. *The complete guide to option pricing formulas*. McGraw-Hill Education - Europe, Jan. 1, 2007. 492 pp. ISBN: 978-0071389976.

[7] J. Hull. *Options, futures, and other derivatives*. Pearson, 2018. XXII, 789. ISBN: 9780134631493.

[8] N. Kunitomo and M. Ikeda. "Pricing Options With Curved Boundaries". In: *Mathematical Finance* 2.4 (Oct. 1992), pp. 275–298. DOI: 10.1111/j.1467-9965.1992.tb00033.x.

[9] B. Mandelbrot and R. Hudson. *The (Mis)Behaviour of Markets: A Fractal View of Risk, Ruin and Reward*. Profile, 2010. ISBN: 9781847651556.

[10] R. U. Seydel. *Tools for Computational Finance*. Springer-Verlag London, 2012. DOI: 10.1007/978-1-4471-2993-6.

[11] S. Theodoridis. "Probability and Stochastic Processes". In: *Machine Learning*. Elsevier, 2020, pp. 19–65. DOI: 10.1016/B978-0-12-818803-3.00011-8.

[12] P. Wilmott, J. N. Dewynne, and S. Howison. *Option pricing : mathematical models and computation*. Oxford Financial Press, 1993. ISBN: 9780952208204.