

Article

# Multi-Asset Barrier Options Pricing by Collocation BEM (with Matlab<sup>®</sup> Code)

Alessandra Aimi <sup>\*,†</sup>  and Chiara Guardasoni <sup>†</sup> 

Department of Mathematical, Physical and Computer Sciences, Parco Area delle Scienze, 53/A,  
43126 Parma, Italy; chiara.guardasoni@unipr.it

\* Correspondence: alessandra.aimi@unipr.it; Tel.: +39-0521-906944

† Members of the INdAM-GNCS Research Group.

**Abstract:** In this paper, we extend the SABO technique (Semi-Analytical method for Barrier Options), based on collocation Boundary Element Method (BEM), to the pricing of Barrier Options with payoff dependent on more than one asset. The efficiency and accuracy already revealed in the case of a single asset is confirmed by the presented numerical results.

**Keywords:** boundary element method; barrier options; multi-asset options; basket options; spread options

**MSC:** 65M38; 91G60; 91G20; 65M80



**Citation:** Aimi, A.; Guardasoni, C. Multi-Asset Barrier Options Pricing by Collocation BEM (with Matlab<sup>®</sup> Code). *Axioms* **2021**, *10*, 301. <https://doi.org/10.3390/axioms10040301>

Academic Editors: Davron Aslonqulovich Juraev, Palle E.T. Jorgensen and Samad Noeiaghdam

Received: 29 September 2021  
Accepted: 10 November 2021  
Published: 12 November 2021

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction to the Differential Model Problem

Partial Differential Equations (PDEs) of Mathematical Physics model a huge variety of real-life problems, from science to engineering. Since the work [1] of F. Black and M. Scholes, equations that model physical phenomena have been reconsidered to interpret financial phenomena. PDEs in space–time variables, modeling the price of the most evolved financial products, need efficient techniques to be numerically solved.

This work investigates the extensions of the so called SABO technique (Semi-Analytical method for Barrier Options), based on collocation Boundary Element Method (BEM) and applied so far for the numerical pricing of barrier options in a one dimensional asset framework [2].

The consideration of two dimensional partial differential problems can be suggested for several reasons: the desire to complicate the model to get closer to reality (such as, for example, introducing the dependence of option value on a stochastic volatility variable [3]) or the evaluation of options that depend not only on the asset value. In this last direction we have already approached Asian options whose payoff depends on the average of asset values [4,5] giving rise to a degenerate PDE based on two independent variables: the asset value and the average.

In this article, the extension is devoted to options whose payoff depends on more than one asset. The consequent differential model, described in the following, is set by a parabolic equation that, with suitable transformations, can be traced back to the heat equation [6].

From the computational point of view, the pricing of multi-asset options is recognized in the literature as a quite difficult issue and difficulties increase with the application of barriers [7]. The problem can be tackled starting from stochastic differential equations by Monte Carlo methods [8,9] or considering the problem in its partial differential formulation [10,11] with some limitations on the number of assets involved.

In principle our strategy can be applied to an undefined number of underlying assets, so theoretically the extension is straightforward, but the choice of approximation technique is crucial to face the *curse of dimensionality*: the application of collocation method ensures

more efficiency than classical domain methods at low dimensions. This will be detailed in Section 4.

We assume the Black–Scholes–Merton scenario for the evaluation of an option  $V(S_1, \dots, S_n, t)$  based on  $n$  assets  $S_1, \dots, S_n$  during the time interval  $[0, T]$ , with no dividends: the behavior of the underlying assets is described by a geometric Brownian motion and  $-1 \leq \rho_{ij} \leq 1$  is the correlation between the two assets  $i$  and  $j$ . The application of  $n$ -dimensional Itô Lemma and no-arbitrage arguments leads to the related backward parabolic differential model problem

$$\frac{\partial V}{\partial t} + \frac{1}{2} \sum_{i,j=1}^n \rho_{ij} \sigma_i \sigma_j S_i S_j \frac{\partial^2 V}{\partial S_i \partial S_j} + r \sum_{i=1}^n S_i \frac{\partial V}{\partial S_i} - rV = 0, \quad (S_1, \dots, S_n, t) \in [\mathbb{R}^+]^n \times [0, T] \quad (1)$$

$$V(S_1, \dots, S_n, T) = V_T(S_1, \dots, S_n), \quad (S_1, \dots, S_n) \in [\mathbb{R}^+]^n$$

where  $r$  is the risk-free interest rate,  $\sigma_i$  is the volatility of underlying asset  $S_i$  and  $V_T$  is the option contract payoff at the expiry  $T$  that may depend on a strike price  $K$  and assume several expressions, for example, looking at call options in [12]:

- $\max(\sum_{i=1}^n c_i S_i - K, 0)$ , for *Basket options* with weights  $c_i$ ;
- for different kinds of *Rainbow options*,
  - $\max(S_1, \dots, S_n)$  *n-color better-of option*
  - $\min(S_1, S_2)$  *two-color worse-of option*
  - $\max(S_2 - S_1, 0)$  *outperformance option*
  - $\max(\min(S_1 - K, \dots, S_n - K), 0)$  *min option*
- $\max(S_1 - S_2 - K, 0)$ , for *spread option*

The discussion in the following sections can be developed in the general dimension  $n$  [13]; however, for the sake of clarity and to simplify numerics, we will detail it considering two underlying assets  $S_1, S_2$  only. Hence, the convenience and reliability of SABO will be highlighted with proofs and numerical examples for a two assets framework, and, in continuity with the previous article in this journal [4], we have inserted in Appendix A a ready to use Matlab® code.

## 2. Integral Representation Formula of the Solution for Options without Barriers

By the Green’s theorem, we prove the integral representation formula for the solution of the differential model problem (1) describing the value of an option  $V(S_1, S_2, t)$  based on two assets  $S_1, S_2$  during the time interval  $[0, T]$ .

We recall the notion of joint transition probability density function  $G(S_1, S_2, t; \tilde{S}_1, \tilde{S}_2, \tilde{t})$  (also known as Green’s or fundamental solution) from the classical theory of PDEs [13]: denoting the space differential operator by

$$\mathcal{L}[V] := \frac{\sigma_1^2 S_1^2}{2} \frac{\partial^2 V}{\partial S_1^2} + r S_1 \frac{\partial V}{\partial S_1} + \frac{\sigma_2^2 S_2^2}{2} \frac{\partial^2 V}{\partial S_2^2} + r S_2 \frac{\partial V}{\partial S_2} + \rho \sigma_1 \sigma_2 S_1 S_2 \frac{\partial^2 V}{\partial S_1 \partial S_2} - rV, \quad (2)$$

$G$  is solution of (1), i.e.,

$$\frac{\partial G}{\partial t}(S_1, S_2, t) + \mathcal{L}[G](S_1, S_2, t) = 0 \quad (3)$$

w.r.t. the first term of variables and of the adjoint model problem (4) and (5) w.r.t. the second term of variables:

$$-\frac{\partial G}{\partial \tilde{t}}(\tilde{S}_1, \tilde{S}_2, \tilde{t}) + \mathcal{L}^*[G](\tilde{S}_1, \tilde{S}_2, \tilde{t}) = 0 \quad (\tilde{S}_1, \tilde{S}_2, \tilde{t}) \in \mathbb{R}^+ \times \mathbb{R}^+ \times (t, +\infty) \quad (4)$$

$$G(S_1, S_2, t; \tilde{S}_1, \tilde{S}_2, \tilde{t}) = \delta(S_1, \tilde{S}_1) \delta(S_2, \tilde{S}_2) \quad (\tilde{S}_1, \tilde{S}_2) \in \mathbb{R}^+ \times \mathbb{R}^+; \quad (5)$$

moreover it is such that

$$(\mathcal{L}[V], G) = (V, \mathcal{L}^*[G]) \quad (6)$$

having set  $(\cdot, \cdot)$  the standard  $L^2$  scalar product. The adjoint operator  $\mathcal{L}^*$  is defined as

$$\begin{aligned} \mathcal{L}^*[G] &:= \frac{\sigma_1^2 \tilde{S}_1^2}{2} \frac{\partial^2 G}{\partial \tilde{S}_1^2} + \frac{\partial G}{\partial \tilde{S}_1} (-r \tilde{S}_1 + 2\sigma_1^2 \tilde{S}_1 + \rho \sigma_1 \sigma_2 \tilde{S}_1) \\ &+ \frac{\sigma_2^2 \tilde{S}_2^2}{2} \frac{\partial^2 G}{\partial \tilde{S}_2^2} + \frac{\partial G}{\partial \tilde{S}_2} (-r \tilde{S}_2 + 2\sigma_2^2 \tilde{S}_2 + \rho \sigma_1 \sigma_2 \tilde{S}_2) \\ &+ \rho \sigma_1 \sigma_2 \tilde{S}_1 \tilde{S}_2 \frac{\partial^2 G}{\partial \tilde{S}_1 \partial \tilde{S}_2} + G(\rho \sigma_1 \sigma_2 - 3r + \sigma_1^2 + \sigma_2^2). \end{aligned} \tag{7}$$

From [14] the fundamental solution is known to be

$$G(S_1, S_2, t; \tilde{S}_1, \tilde{S}_2, \tilde{t}) = \frac{e^{-r(\tilde{t}-t)} \exp\left(-\frac{\alpha' \Sigma^{-1} \alpha}{2}\right)}{2\pi(\tilde{t}-t) \sigma_1 \sigma_2 \tilde{S}_1 \tilde{S}_2 \sqrt{\det \Sigma}} \tag{8}$$

with

$$\alpha_i = \frac{\log \frac{S_i}{\tilde{S}_i} + \left(r - \frac{\sigma_i^2}{2}\right)(\tilde{t}-t)}{\sigma_i \sqrt{\tilde{t}-t}}, \quad i = 1, 2$$

the elements of the array  $\alpha$  and

$$\Sigma = \begin{pmatrix} 1 & \rho_{12} \\ \rho_{12} & 1 \end{pmatrix}.$$

the correlation matrix.

**Proposition 1.** *The fundamental solution (8) of (3)–(5) verifies,  $\forall \tilde{t}, t \in [0, T]$ , the following identities*

$$\mathcal{I}_1 := \int_{\mathbb{R}^+ \times \mathbb{R}^+} G(S_1, S_2, t; \tilde{S}_1, \tilde{S}_2, \tilde{t}) d\tilde{S}_1 d\tilde{S}_2 = e^{-r(\tilde{t}-t)} \tag{9}$$

$$\mathcal{I}_2 := \int_{\mathbb{R}^+ \times \mathbb{R}^+} G(S_1, S_2, t; \tilde{S}_1, \tilde{S}_2, \tilde{t}) dS_1 dS_2 = e^{-(\tilde{t}-t)(3r - \sigma_1 \sigma_2 \rho_{12} - \sigma_1^2 - \sigma_2^2)} \tag{10}$$

**Proof.** From (8),

$$G(S_1, S_2, t; \tilde{S}_1, \tilde{S}_2, \tilde{t}) = \frac{e^{-r(\tilde{t}-t)} \exp\left(-\frac{\alpha_1^2 + \alpha_2^2 - 2\alpha_1 \alpha_2 \rho_{12}}{2(1-\rho_{12}^2)}\right)}{2\pi(\tilde{t}-t) \sigma_1 \sigma_2 \tilde{S}_1 \tilde{S}_2 \sqrt{1-\rho_{12}^2}}$$

hence

$$\mathcal{I}_1 = \frac{e^{-r(\tilde{t}-t)}}{2\pi(\tilde{t}-t) \sigma_1 \sigma_2 \sqrt{1-\rho_{12}^2}} \int_{\mathbb{R}^+} \frac{\exp\left(-\frac{\alpha_1^2}{2(1-\rho_{12}^2)}\right)}{\tilde{S}_1} d\tilde{S}_1 \int_{\mathbb{R}^+} \frac{\exp\left(-\frac{\alpha_2^2 - 2\alpha_1 \alpha_2 \rho_{12}}{2(1-\rho_{12}^2)}\right)}{\tilde{S}_2} d\tilde{S}_2$$

by the changes of variables  $\xi_i := \log(\tilde{S}_i)$  and exploiting the property  $\int_0^{+\infty} e^{-\xi^2} d\xi = \sqrt{\pi}/2$ , we obtain

$$\begin{aligned} &= \frac{e^{-r(\tilde{t}-t)}}{2\pi(\tilde{t}-t) \sigma_1 \sigma_2 \sqrt{1-\rho_{12}^2}} \int_{\mathbb{R}^+} e^{-\frac{\alpha_1^2}{2}} \sqrt{2\pi \sigma_2^2 (\tilde{t}-t) (1-\rho_{12}^2)} d\xi_1 \\ &= \frac{e^{-r(\tilde{t}-t)}}{\sqrt{2\pi(\tilde{t}-t)} \sigma_1} \int_{\mathbb{R}^+} e^{-\frac{\alpha_1^2}{2}} d\xi_1 = e^{-r(\tilde{t}-t)}. \end{aligned}$$

Analogously

$$\begin{aligned} \mathcal{I}_2 &= \frac{e^{-r(\bar{t}-t)}}{2\pi(\bar{t}-t)\sigma_1\sigma_2\tilde{S}_1\tilde{S}_2\sqrt{1-\rho_{12}^2}} \int_{\mathbb{R}^+} e^{-\frac{\alpha_1^2}{2(1-\rho_{12}^2)}} dS_1 \int_{\mathbb{R}^+} e^{-\frac{\alpha_2^2-2\alpha_1\alpha_2\rho_{12}}{2(1-\rho_{12}^2)}} dS_2 \\ &= \frac{e^{(\bar{t}-t)(-2r+\sigma_2^2-\frac{\rho_{12}^2\sigma_2^2}{2})}}{\sqrt{2\pi(\bar{t}-t)}\sigma_1\tilde{S}_1} \int_{\mathbb{R}^+} e^{-\frac{\alpha_1^2}{2}+\alpha_1\rho_{12}(\bar{t}-t)\sigma_2} dS_1 = e^{(\bar{t}-t)(-3r+\sigma_2^2+\sigma_1^2+\sigma_1\sigma_2\rho_{12})} \end{aligned}$$

□

**Remark 1.** Note that  $\mathcal{I}_1$  and  $\mathcal{I}_2$  turn out to coincide with the coefficients of the changes of variables  $\tilde{G} = e^{r(\bar{t}-t)}G$  and  $\tilde{G} = e^{(3r-\sigma_1\sigma_2\rho_{12}-\sigma_1^2-\sigma_2^2)(\bar{t}-t)}G$  allowing to obtain (3) and (4), respectively, without the last term.

With the knowledge of the fundamental solution, we will derive, by mathematical analysis tools, the following integral representation formula:

**Proposition 2.** The Feynman–Kac formula

$$V(S_1, S_2, t) = \int_{\mathbb{R}^+ \times \mathbb{R}^+} V(\tilde{S}_1, \tilde{S}_2, T)G(S_1, S_2, t; \tilde{S}_1, \tilde{S}_2, T)d\tilde{S}_1d\tilde{S}_2 \tag{11}$$

giving the option value as expected value of the payoff function, holds.

**Proof.** Multiply (1) by  $G$  and integrate in time-space domain

$$\begin{aligned} 0 &= \int_t^T \int_{\mathbb{R}^+ \times \mathbb{R}^+} \frac{\partial V}{\partial \bar{t}}(\tilde{S}_1, \tilde{S}_2, \bar{t})G(S_1, S_2, t; \tilde{S}_1, \tilde{S}_2, \bar{t})d\bar{t}d\tilde{S}_1d\tilde{S}_2 \\ &+ \int_t^T \int_{\mathbb{R}^+ \times \mathbb{R}^+} \mathcal{L}[V](\tilde{S}_1, \tilde{S}_2, \bar{t})G(S_1, S_2, t; \tilde{S}_1, \tilde{S}_2, \bar{t})d\bar{t}d\tilde{S}_1d\tilde{S}_2 \end{aligned}$$

then, integrating by parts in time and in space and applying (6), one obtains

$$\begin{aligned} &= \int_{\mathbb{R}^+ \times \mathbb{R}^+} V(\tilde{S}_1, \tilde{S}_2, T)G(S_1, S_2, t; \tilde{S}_1, \tilde{S}_2, T)d\tilde{S}_1d\tilde{S}_2 \\ &- \int_{\mathbb{R}^+ \times \mathbb{R}^+} V(\tilde{S}_1, \tilde{S}_2, t)G(S_1, S_2, t; \tilde{S}_1, \tilde{S}_2, t)d\tilde{S}_1d\tilde{S}_2 \\ &- \int_t^T \int_{\mathbb{R}^+ \times \mathbb{R}^+} V(\tilde{S}_1, \tilde{S}_2, \bar{t})\frac{\partial G}{\partial \bar{t}}(S_1, S_2, t; \tilde{S}_1, \tilde{S}_2, \bar{t})d\bar{t}d\tilde{S}_1d\tilde{S}_2 \\ &+ \int_t^T \int_{\mathbb{R}^+ \times \mathbb{R}^+} V(\tilde{S}_1, \tilde{S}_2, \bar{t})\mathcal{L}^*[G](S_1, S_2, t; \tilde{S}_1, \tilde{S}_2, \bar{t})d\bar{t}d\tilde{S}_1d\tilde{S}_2 \\ &= \int_{\mathbb{R}^+ \times \mathbb{R}^+} V(\tilde{S}_1, \tilde{S}_2, T)G(S_1, S_2, t; \tilde{S}_1, \tilde{S}_2, T)d\tilde{S}_1d\tilde{S}_2 \\ &- \int_{\mathbb{R}^+ \times \mathbb{R}^+} V(\tilde{S}_1, \tilde{S}_2, t)\delta(S_1, \tilde{S}_1)\delta(S_2, \tilde{S}_2)d\tilde{S}_1d\tilde{S}_2 \\ &- \int_t^T \int_{\mathbb{R}^+ \times \mathbb{R}^+} V(\tilde{S}_1, \tilde{S}_2, \bar{t})\left(\frac{\partial G}{\partial \bar{t}} - \mathcal{L}^*[G]\right)(S_1, S_2, t; \tilde{S}_1, \tilde{S}_2, \bar{t})d\bar{t}d\tilde{S}_1d\tilde{S}_2 \end{aligned}$$

and then, thanks to property (6), these steps lead to the Feynman-Kac formula. □

### 3. Integral Representation Formula of the Solution for Barrier Options

If we consider barrier options, sometimes analytical solutions are known (some examples are collected in [15]) but sometimes not. For example, in [12], the author considers the case of a European-style two-assets-basket double-barrier call option with payoff

$$V(S_1, S_2, T) = \max(S_1 + S_2 - K, 0) \tag{12}$$

and two knock-out barrier conditions

$$V(S_1, S_2, T) = 0 \quad S_1 + S_2 \leq B_1 \tag{13}$$

$$V(S_1, S_2, T) = 0 \quad S_1 + S_2 \geq B_2 \tag{14}$$

with down-and-out barrier  $B_1$  and up-and-out barrier  $B_2$ . Then, the resulting domain  $\Omega$  is partitioned as shown in Figure 1 in order to approximate the option value by a Finite Element Method.

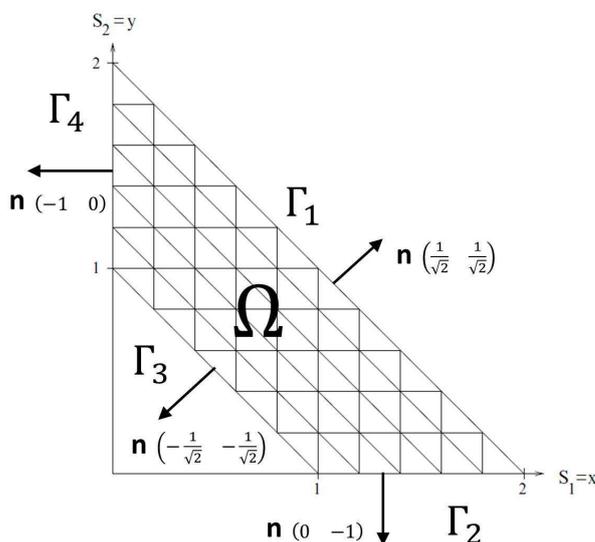


Figure 1. Triangulation of domain  $\Omega$  in [12].

However, if we are interested in the computation of the option value only at a desired point  $(S_1, S_2)$  of the domain, it may certainly be convenient to apply the strategy already tested for various kind of options (based on one asset only) under several dynamics [2–5,16,17] and that we called SABO.

The method is based on a new analytical representation formula whose proof relies on Green’s Theorem and retraces the proof of Feynman–Kac formula.

**Proposition 3.** *The solution of problem (1), (13) and (14),  $\forall (S_1, S_2, t) \in \Omega \subseteq \mathbb{R}^2 \times [0, T)$ , can be expressed by the following integral representation formula:*

$$V(S_1, S_2, t) = \int_{\Omega} V(\tilde{S}_1, \tilde{S}_2, T)G(S_1, S_2, t; \tilde{S}_1, \tilde{S}_2, T)d\tilde{S}_1d\tilde{S}_2 \tag{15}$$

$$+ \int_t^T \int_{\partial\Omega} \varphi(\tilde{S}_1, \tilde{S}_2, \tilde{t}) d\tilde{t}d\tilde{S}_1d\tilde{S}_2$$

with the functional  $\varphi$  defined in (17) below.

**Proof.** Multiply (1) by  $G$  and integrate in time and in the domain  $\Omega$

$$0 = \int_t^T \int_{\Omega} \frac{\partial V}{\partial \tilde{t}}(\tilde{S}_1, \tilde{S}_2, \tilde{t}) G(S_1, S_2, t; \tilde{S}_1, \tilde{S}_2, \tilde{t}) d\tilde{t} d\tilde{S}_1 d\tilde{S}_2 + \int_t^T \int_{\Omega} \mathcal{L}[V](\tilde{S}_1, \tilde{S}_2, \tilde{t}) G(S_1, S_2, t; \tilde{S}_1, \tilde{S}_2, \tilde{t}) d\tilde{t} d\tilde{S}_1 d\tilde{S}_2$$

then, integrating by parts in time and applying Green’s Theorem and (6), one obtains

$$\begin{aligned} &= \int_{\Omega} V(\tilde{S}_1, \tilde{S}_2, T) G(S_1, S_2, t; \tilde{S}_1, \tilde{S}_2, T) d\tilde{S}_1 d\tilde{S}_2 \\ &- \int_{\Omega} V(\tilde{S}_1, \tilde{S}_2, t) G(S_1, S_2, t; \tilde{S}_1, \tilde{S}_2, t) d\tilde{S}_1 d\tilde{S}_2 \\ &- \int_t^T \int_{\Omega} V(\tilde{S}_1, \tilde{S}_2, \tilde{t}) \frac{\partial G}{\partial \tilde{t}}(S_1, S_2, t; \tilde{S}_1, \tilde{S}_2, \tilde{t}) d\tilde{t} d\tilde{S}_1 d\tilde{S}_2 \\ &+ \int_t^T \int_{\Omega} V(\tilde{S}_1, \tilde{S}_2, \tilde{t}) \mathcal{L}^*[G](S_1, S_2, t; \tilde{S}_1, \tilde{S}_2, \tilde{t}) d\tilde{t} d\tilde{S}_1 d\tilde{S}_2 \\ &+ \int_t^T \int_{\partial\Omega} \mathbf{p}(S_1, S_2, t; \tilde{S}_1, \tilde{S}_2, \tilde{t}) \cdot \mathbf{n} d\tilde{t} d\tilde{S}_1 d\tilde{S}_2 \end{aligned}$$

having defined  $\partial\Omega$  as the boundary of the domain,  $\mathbf{n}$  as its unit normal vector outwardly directed and  $\mathbf{p} = (p_1, p_2)$  as

$$\begin{aligned} p_1 &= \frac{1}{2} \sigma_1^2 \tilde{S}_1^2 \left( G \frac{\partial V}{\partial \tilde{S}_1} - V \frac{\partial G}{\partial \tilde{S}_1} \right) + \frac{1}{2} \rho \sigma_1 \sigma_2 \tilde{S}_1 \tilde{S}_2 \left( G \frac{\partial V}{\partial \tilde{S}_2} - V \frac{\partial G}{\partial \tilde{S}_2} \right) \\ &- VG \left( \sigma_1^2 \tilde{S}_1 + \frac{1}{2} \rho \sigma_1 \sigma_2 \tilde{S}_1 - r \tilde{S}_1 \right) \\ p_2 &= \frac{1}{2} \sigma_2^2 \tilde{S}_2^2 \left( G \frac{\partial V}{\partial \tilde{S}_2} - V \frac{\partial G}{\partial \tilde{S}_2} \right) + \frac{1}{2} \rho \sigma_1 \sigma_2 \tilde{S}_1 \tilde{S}_2 \left( G \frac{\partial V}{\partial \tilde{S}_1} - V \frac{\partial G}{\partial \tilde{S}_1} \right) \\ &- VG \left( \sigma_2^2 \tilde{S}_2 + \frac{1}{2} \rho \sigma_1 \sigma_2 \tilde{S}_2 - r \tilde{S}_2 \right) \end{aligned} \tag{16}$$

Thus we can conclude that

$$\begin{aligned} 0 &= \int_{\Omega} V(\tilde{S}_1, \tilde{S}_2, T) G(S_1, S_2, t; \tilde{S}_1, \tilde{S}_2, T) d\tilde{S}_1 d\tilde{S}_2 \\ &- \int_{\Omega} V(\tilde{S}_1, \tilde{S}_2, t) \delta(S_1, \tilde{S}_1) \delta(S_2, \tilde{S}_2) d\tilde{S}_1 d\tilde{S}_2 \\ &- \int_t^T \int_{\Omega} V(\tilde{S}_1, \tilde{S}_2, \tilde{t}) \left( \frac{\partial G}{\partial \tilde{t}} - \mathcal{L}^*[G] \right) (S_1, S_2, t; \tilde{S}_1, \tilde{S}_2, \tilde{t}) d\tilde{t} d\tilde{S}_1 d\tilde{S}_2 \\ &+ \int_t^T \int_{\partial\Omega} \varphi(S_1, S_2, t; \tilde{S}_1, \tilde{S}_2, \tilde{t}) d\tilde{t} d\tilde{S}_1 d\tilde{S}_2 \end{aligned}$$

with

$$\varphi(S_1, S_2, t; \tilde{S}_1, \tilde{S}_2, \tilde{t}) = \mathbf{p}(S_1, S_2, t; \tilde{S}_1, \tilde{S}_2, \tilde{t}) \cdot \mathbf{n} \tag{17}$$

□

**Remark 2.** The representation formula, and SABO in general, can be naturally adapted to barrier options configurations different from the proposed example. Note also that it is not really important to know exactly the expression of the functional  $\varphi$  because, as it depends on the unknown solution  $V$ , it is unknown in its turn and therefore it will be numerically determined.

**Remark 3.** The representation formula can be analytically derived in time or in space to obtain Greeks functions that can be straightforwardly evaluated by SABO without the need of evaluating option values (look at [2]).

In the example (12)–(14), due to the configuration of the boundary  $\partial\Omega = \Gamma_1 \cup \Gamma_2 \cup \Gamma_3 \cup \Gamma_4$  (see Figure 1), the integral of  $\varphi$  reduces to

$$\int_t^T \int_{\partial\Omega} \varphi(S_1, S_2, t; \tilde{S}_1, \tilde{S}_2, \tilde{t}) \, d\tilde{t} d\tilde{S}_1 d\tilde{S}_2 = \int_t^T \int_{\Gamma_1 \cup \Gamma_3} \varphi(S_1, S_2, t; \tilde{S}_1, \tilde{S}_2, \tilde{t}) \, d\tilde{t} d\tilde{S}_1 d\tilde{S}_2$$

because on  $\Gamma_2 := \{(S_1, 0) : B_1 \leq S_1 \leq B_2\}$  the first component of the normal vector is null and every term of  $p_2$  has the factor  $S_2$ , trivial on  $\Gamma_2$ ; on  $\Gamma_4 := \{(0, S_2) : B_1 \leq S_2 \leq B_2\}$  the second component of the normal vector is null and every term of  $p_1$  has the factor  $S_1$ , trivial on  $\Gamma_4$ .

Moreover, considering double knock-out barriers on  $\Gamma_1 := \{(S_1, S_2) : S_1 + S_2 = B_2\}$  and on  $\Gamma_3 := \{(S_1, S_2) : S_1 + S_2 = B_1\}$ , the option value  $V$  is equal to 0, justifying the representation formula:  $\forall (S_1, S_2, t) \in \Omega \times [0, T)$

$$\begin{aligned} V(S_1, S_2, t) &= \int_{\Omega} V(\tilde{S}_1, \tilde{S}_2, T) G(S_1, S_2, t; \tilde{S}_1, \tilde{S}_2, T) d\tilde{S}_1 d\tilde{S}_2 \\ &+ \int_t^T \int_{\Gamma_1 \cup \Gamma_3} G(S_1, S_2, t; \tilde{S}_1, \tilde{S}_2, \tilde{t}) \phi(\tilde{S}_1, \tilde{S}_2, \tilde{t}) \, d\tilde{t} d\tilde{S}_1 d\tilde{S}_2 \end{aligned} \tag{18}$$

with  $\phi$  depending only on  $(\tilde{S}_1, \tilde{S}_2, \tilde{t}) \in \Gamma_1 \cup \Gamma_3 \times [0, T)$ , i.e.,

$$\phi(\tilde{S}_1, \tilde{S}_2, \tilde{t}) = \left( \frac{1}{2} \sigma_1^2 \tilde{S}_1^2 \frac{\partial V}{\partial \tilde{S}_1} + \frac{1}{2} \rho \sigma_1 \sigma_2 \tilde{S}_1 \tilde{S}_2 \frac{\partial V}{\partial \tilde{S}_2} \right) n_1 + \left( \frac{1}{2} \sigma_2^2 \tilde{S}_2^2 \frac{\partial V}{\partial \tilde{S}_2} + \frac{1}{2} \rho \sigma_1 \sigma_2 \tilde{S}_1 \tilde{S}_2 \frac{\partial V}{\partial \tilde{S}_1} \right) n_2.$$

#### 4. Boundary Integral Equation

The representation formula (18) of  $V$  holds in the whole domain and can be used once the unknown density  $\phi$  is recovered. To this aim, the idea is to consider the application of the representation formula at the barriers, where  $\forall (S_1, S_2, t) \in \Gamma_1 \cup \Gamma_3 \times [0, T)$  the option value is trivial, i.e.,  $V(S_1, S_2, t) = 0$  and therefore

$$\begin{aligned} \int_t^T \int_{\Gamma_1 \cup \Gamma_3} G(S_1, S_2, t; \tilde{S}_1, \tilde{S}_2, \tilde{t}) \phi(\tilde{S}_1, \tilde{S}_2, \tilde{t}) \, d\tilde{t} d\tilde{S}_1 d\tilde{S}_2 &= \\ - \int_{\Omega} V(\tilde{S}_1, \tilde{S}_2, T) G(S_1, S_2, t; \tilde{S}_1, \tilde{S}_2, T) d\tilde{S}_1 d\tilde{S}_2. \end{aligned} \tag{19}$$

This Boundary Integral Equation (BIE) in the unknown  $\phi$  is a Volterra integral equation of first kind and can be numerically solved as described in the following section.

**Remark 4.** This strategy can be extended to higher dimensions, denoting by  $\Gamma$  the hyperplane defined by the barrier conditions. The representation formula still holds:  $\forall (S_1, \dots, S_n, t) \in \Omega \times [0, T)$

$$\begin{aligned} V(S_1, \dots, S_n, t) &= \int_{\Omega} V(\tilde{S}_1, \dots, \tilde{S}_n, T) G(S_1, \dots, S_n, t; \tilde{S}_1, \dots, \tilde{S}_n, T) d\tilde{S}_1 \dots d\tilde{S}_n \\ &+ \int_t^T \int_{\Gamma} G(S_1, \dots, S_n, t; \tilde{S}_1, \dots, \tilde{S}_n, \tilde{t}) \phi(\tilde{S}_1, \dots, \tilde{S}_n, \tilde{t}) \, d\tilde{t} d\tilde{S}_1 \dots d\tilde{S}_n \end{aligned} \tag{20}$$

together with the BIE with knock-out condition:  $\forall (S_1, S_2, t) \in \Gamma \times [0, T)$

$$\begin{aligned} \int_t^T \int_{\Gamma} G(S_1, \dots, S_n, t; \tilde{S}_1, \dots, \tilde{S}_n, \tilde{t}) \phi(\tilde{S}_1, \dots, \tilde{S}_n, \tilde{t}) \, d\tilde{t} d\tilde{S}_1, \dots, d\tilde{S}_n &= \\ - \int_{\Omega} V(\tilde{S}_1, \dots, \tilde{S}_n, T) G(S_1, \dots, S_n, t; \tilde{S}_1, \dots, \tilde{S}_n, T) d\tilde{S}_1 d\tilde{S}_2 \end{aligned} \tag{21}$$

and with the general expression of fundamental solution  $G$  written in [14].

**Remark 5.** If the asset domain is either a 2 or 3 or 4 dimensional domain, to solve (21) means to reduce the dimensionality of the problem by one. In this case, the collocation method can be implemented, as proposed in Section 5 (introducing suitable mesh triangulations), since finite

element discretization can be easily applied up to three dimensions with presumably greater accuracy w.r.t. Monte Carlo simulations. However, of course, the collocation method suffers the curse of dimensionality. Therefore, first of all, SABO in this framework, is suggested as an alternative to deterministic methods (finite difference methods and finite element methods) for its higher efficiency.

With the increasing of the dimension  $n > 4$ , (21) is still valid and very general, not requiring special conditions as for example in [18], so the strategy of solving (21) can be anyway taken into account. An idea could be to involve Monte Carlo method, not as a method for path simulation [9], but considered as a quadrature method [19] directly applied to the integral terms in (21). Then unknown  $\phi$  might be represented by linear combination of radial basis functions around some nodes spread out in the  $n - 1$ -dimensional hyperplane  $\Gamma$ . This will be the subject of future investigation.

### 5. Numerical Approximation by Collocation BEM

We introduce

$$\tilde{\phi}(S_1, S_2, t) := \sum_{n=1}^{N_{\Delta t}} \sum_{m=1}^{M_{\Delta S}} \alpha_{nm} \bar{\phi}_m(S_1, S_2) \hat{\phi}_n(t) \tag{22}$$

for the numerical approximation of  $\phi$  by:

- piecewise constant functions in time

$$\hat{\phi}_n(t) := H[t - t_{n-1}] - H[t - t_n] \quad n = 1, \dots, N_{\Delta t}$$

defined by the Heaviside functions  $H[\cdot]$  on a uniform decomposition of the time interval  $[0, T]$ :

$$\Delta t = \frac{T}{N_{\Delta t}}, \quad N_{\Delta t} \in \mathbb{N}^+ \quad t_n = nN_{\Delta t}, \quad n = 0, \dots, N_{\Delta t},$$

- piecewise constant functions in space  $\bar{\phi}_m(S) = \bar{\phi}_m(S_1, S_2)$ ,  $m = 1, \dots, \bar{M}_{\Delta S}, \bar{M}_{\Delta S} + 1, \dots, M_{\Delta S}$  defined on a decomposition  $\mathcal{T}_1 = \{e_1, \dots, e_{\bar{M}_{\Delta S}}\}$  on  $\Gamma_1$  and  $\mathcal{T}_2 = \{e_{\bar{M}_{\Delta S}+1}, \dots, e_{M_{\Delta S}}\}$  on  $\Gamma_2$  constituted by  $M_{\Delta S}$  segments such that  $\text{length}(e_i) \leq \Delta S$  with  $e_i \cap e_j = \emptyset$  if  $i \neq j$ .

The related unknown array

$$\alpha = \left( \alpha^{(1)} \alpha^{(1)} \alpha^{(2)} \dots \alpha^{(N_{\Delta t})} \right)^\top \quad \alpha^{(\ell)} = (\alpha_{\ell 1}, \dots, \alpha_{\ell M_{\Delta S}})^\top, \quad \ell = 1, \dots, N_{\Delta t}$$

can be computed by solving a linear system

$$\mathbb{M}\alpha = \beta, \tag{23}$$

of  $N_{\Delta t} \times M_{\Delta S}$  equations obtained by collocating the BIE (19) at the midpoints of time decomposition interval

$$\bar{t}_n = \frac{t_{n-1} + t_n}{2}, \quad n = 1, \dots, N_{\Delta t} \tag{24}$$

and at the midpoints of space decomposition segments

$$\bar{S}_m = (\bar{S}_1^{(m)}, \bar{S}_2^{(m)}), \quad m = 1, \dots, M_{\Delta S}. \tag{25}$$

Note that the matrix  $\mathbb{M}$  has a block upper triangular Toeplitz structure

$$\begin{pmatrix} \mathbb{M}^{(1)} & \mathbb{M}^{(2)} & \mathbb{M}^{(3)} & \dots & \mathbb{M}^{(N_{\Delta t})} \\ 0 & \mathbb{M}^{(1)} & \mathbb{M}^{(2)} & \dots & \mathbb{M}^{(N_{\Delta t}-1)} \\ 0 & 0 & \mathbb{M}^{(1)} & \dots & \mathbb{M}^{(N_{\Delta t}-2)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & \mathbb{M}^{(1)} \end{pmatrix} \tag{26}$$

and this is due to the form of the fundamental solution (8): the starting PDE Equation (1) has coefficients independent of time implying that the fundamental solution depends on time only through the difference  $\tilde{t} - t$  and, by consequence, each block depends only on  $t_n - \tilde{t}_n$ . From the computational point of view, this means notable computational and memory savings because only the last column of blocks needs to be computed and the linear system (23) can be solved by block-backward substitution

$$\begin{cases} \mathbf{z}^{(\ell)} = \boldsymbol{\beta}^{(\ell)} - \sum_{j=2}^{N_{\Delta t} - \ell + 1} \mathbb{M}^{(j)} \boldsymbol{\alpha}^{(\ell+j-1)} \\ \mathbb{M}^{(1)} \boldsymbol{\alpha}^{(\ell)} = \mathbf{z}^{(\ell)}. \end{cases} \quad \ell = N_{\Delta t}, \dots, 1$$

where only the non singular block  $\mathbb{M}^{(1)}$  needs to be inverted.

We can get the expression of a general entry of the matrix block  $\mathbb{M}^{(\ell)}$ ,  $\ell = 1, \dots, N_{\Delta t}$ , introducing, in the lhs of (19), a space-time piecewise constant approximation (22) and collocation points (24) and (25):  $\forall \bar{m}, m = 1 \dots, M_{\Delta S}, \quad \forall \bar{n} = 1 \dots, N_{\Delta t}, \forall n = \bar{n} \dots, N_{\Delta t}$

$$\begin{aligned} \mathbb{M}_{\bar{m}m}^{(\ell)} &= \mathbb{M}_{\bar{m}m}^{\bar{n}n} = \mathbb{M}_{\bar{m}m}^{(n-\bar{n}+1)} = \int_{\tilde{t}_{\bar{n}}}^T \int_{e_m} G(\bar{S}_1^{(\bar{m})}, \bar{S}_2^{(\bar{m})}, \tilde{t}_{\bar{n}}; \tilde{S}_1, \tilde{S}_2, \tilde{t}) \bar{\phi}_m(\tilde{S}_1, \tilde{S}_2) \hat{\phi}_n(\tilde{t}) \, d\tilde{t} \, d\tilde{S}_1 \, d\tilde{S}_2 \\ &= \int_{\max(t_{n-1}, \tilde{t}_{\bar{n}})}^{t_n} \int_{e_m} G(\bar{S}_1^{(\bar{m})}, \bar{S}_2^{(\bar{m})}, \tilde{t}_n; \tilde{S}_1, \tilde{S}_2, \tilde{t}) \, d\tilde{t} \, d\tilde{S}_1 \, d\tilde{S}_2 \end{aligned} \quad (27)$$

and, collocating the rhs of (19) in the same way, we can write

$$\beta_{\bar{m}}^{\ell} = \beta_{\bar{m}}^{\bar{n}} = - \int_{\Omega} V(\tilde{S}_1, \tilde{S}_2, T) G(\bar{S}_1^{(\bar{m})}, \bar{S}_2^{(\bar{m})}, \tilde{t}_{\bar{n}}; \tilde{S}_1, \tilde{S}_2, T) \, d\tilde{S}. \quad (28)$$

Once the array  $\boldsymbol{\alpha}$  of coefficients in (22) are computed by solving the linear system (23), the unknown functional  $\phi$  in (18) can be replaced by its approximation  $\tilde{\phi}$ , in order to get the approximation of the option price, at the desired time instant and assets values, without the need of introducing grids or triangulation over the domain  $\Omega$  as in Figure 1.

Note that the evaluation of system entries can be simply performed by Matlab® “integral2” function as it presents only little troubles: the degeneration of the fundamental solution in time towards the Dirac delta distribution when  $t \rightarrow \tilde{t}$  (look at (5)) and possible non-smoothness of the payoff function like, for example, in the case of Basket options.

### 6. Numerical Results and Discussion

- Consider the problem (1), (12), (13) and (14) of evaluating a double knock-out basket call option based on two assets

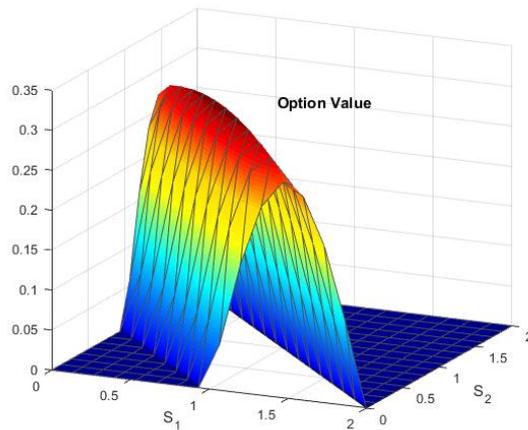
$$V(S_1, S_2, T) = \max(S_1 + S_2 - K, 0)$$

with data suggested in [12] and listed in Table 1. The domain is as depicted in Figure 1. Numerical results have been obtained by the code inserted in the Appendix A.

**Table 1.** Double knock-out basket call option data.

$K$	$T$	$r$	$\sigma_1$	$\sigma_2$	$\rho$	$B_1$	$B_2$
1	1	0.05	0.25	0.25	0.7	1	2

The approximated solution, represented in Figure 2, is evaluated over a rectangular grid of points, vertices of  $16^2$  simplices that subdivide the square  $[0, 2]^2$ . It is obtained by SABO with  $\Delta S = 0.125\sqrt{2}$  and  $\Delta t = 0.1$ . The behavior of the solution is in good agreement with that found in [12] (“Courtesy of A. Kvetnaia”), there approximated by Finite Element Method (FEM) after having transformed Equation (1) in a divergence-free form, not necessary or useful to apply SABO.

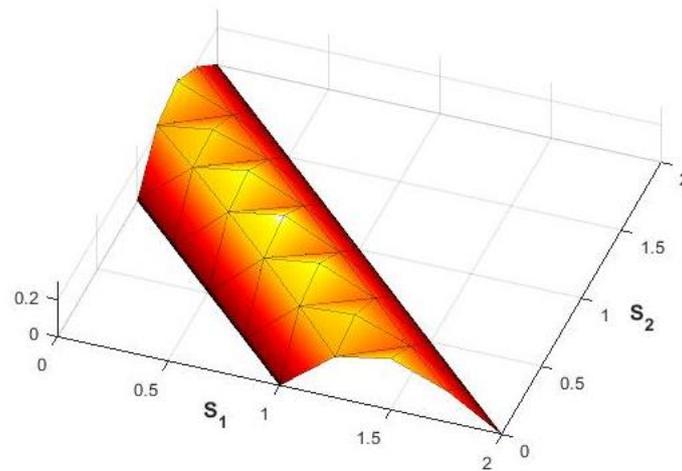


**Figure 2.** SABO approximation of a double knock-out basket call option value  $V(S_1, S_2, 0)$  with data suggested in Table 1.

The greater accuracy, implying greater velocity, of SABO w.r.t. Finite Differences and Monte Carlo Methods has been already highlighted in our papers about barrier options, see for example [3,20]. Here, in Table 2, we compare the CPU time of computation (Laptop computer: CPU Intel i5, 4 Gb RAM.) of SABO w.r.t. FEM as applied in [12], i.e., linear finite elements in space coupled with Crank–Nicolson scheme in time: the chosen reference value is  $V(0.5, 1, 0) = 0.306264$ , obtained by stressing the discretization parameters in our SABO code ( $M_{\Delta S} = N_{\Delta t} = 30$ , tolerance on integrals computation equal to  $10^{-12}$ ) and both methods converge towards it. The starting space mesh for FEM, that will be denoted by  $R_0$  from now on, is made by 48 uniform triangles as depicted in Figure 3, then any successive refinement,  $R_i$   $i = 1, 2, 3$ , is obtained by dividing each triangle in four uniform triangles (by connecting the midpoints of the edges). Despite not having implemented a parallel code for the computation of independent blocks in (26), SABO allows to achieve the same accuracy of FEM with a computation time of at least one lower order of magnitude.

**Table 2.** Comparison between FEM and SABO: option value obtained at  $S_1 = 0.5, S_2 = 1, t = 0$  (upper tables) and CPU time of computation in seconds (lower tables). Reference value: 0.306264.

		SABO				FEM			
$M_{\Delta S}$	$N_{\Delta t}$	2	4	8	16	$N_{\Delta t}$	25	50	100
		2	0.304101	0.305635	0.305527	0.305501	$R_0$	0.293627	0.293663
	4	0.305242	0.306735	0.306631	0.306586	$R_1$	0.304109	0.304129	0.304134
	8	0.304903	0.306424	0.306320	0.306274	$R_2$	0.305875	0.305891	0.305894
	16	0.304916	0.306429	0.306329	0.306282	$R_3$	0.306171	0.306185	0.306188
						$R_4$	0.306229	0.306243	0.306246
		CPU time SABO				CPU time FEM			
$M_{\Delta S}$	$N_{\Delta t}$	2	4	8	16	$N_{\Delta t}$	25	50	100
	2	$2.1 \times 10^{-1}$	$5.8 \times 10^{-1}$	$8.6 \times 10^{-1}$	$1.1 \times 10^0$	$R_0$	$7.2 \times 10^{-1}$	$1.4 \times 10^0$	$3.4 \times 10^0$
	4	$5.1 \times 10^{-1}$	$7.8 \times 10^{-1}$	$1.2 \times 10^0$	$2.1 \times 10^0$	$R_1$	$4.3 \times 10^0$	$8.3 \times 10^0$	$1.5 \times 10^1$
	8	$9.2 \times 10^{-1}$	$1.5 \times 10^0$	$2.5 \times 10^0$	$4.5 \times 10^0$	$R_2$	$8.1 \times 10^0$	$1.3 \times 10^1$	$2.2 \times 10^1$
	16	$2.1 \times 10^0$	$4.1 \times 10^0$	$6.8 \times 10^0$	$1.3 \times 10^0$	$R_3$	$3.0 \times 10^1$	$3.6 \times 10^1$	$4.8 \times 10^1$
						$R_4$	$6.4 \times 10^2$	$6.7 \times 10^2$	$7.1 \times 10^2$



**Figure 3.** FEM approximation of a double knock-out basket call option value  $V(S_1, S_2, 0)$  with data suggested in Table 1, mesh  $R_0$  and  $\Delta t = 0.02$ .

In order to quantitatively check the reliability of numerical results, we show in Figure 4 that the approximated solution evaluated inside the domain but moving towards boundary  $\Gamma_4$  (or equivalently  $\Gamma_2$ ) converges to the exact solution of a double knock-out call option with the same financial parameters but based on one asset, whose explicit expression  $C(S_1, t)$  can be found in [21] and is here reported for reader’s convenience:

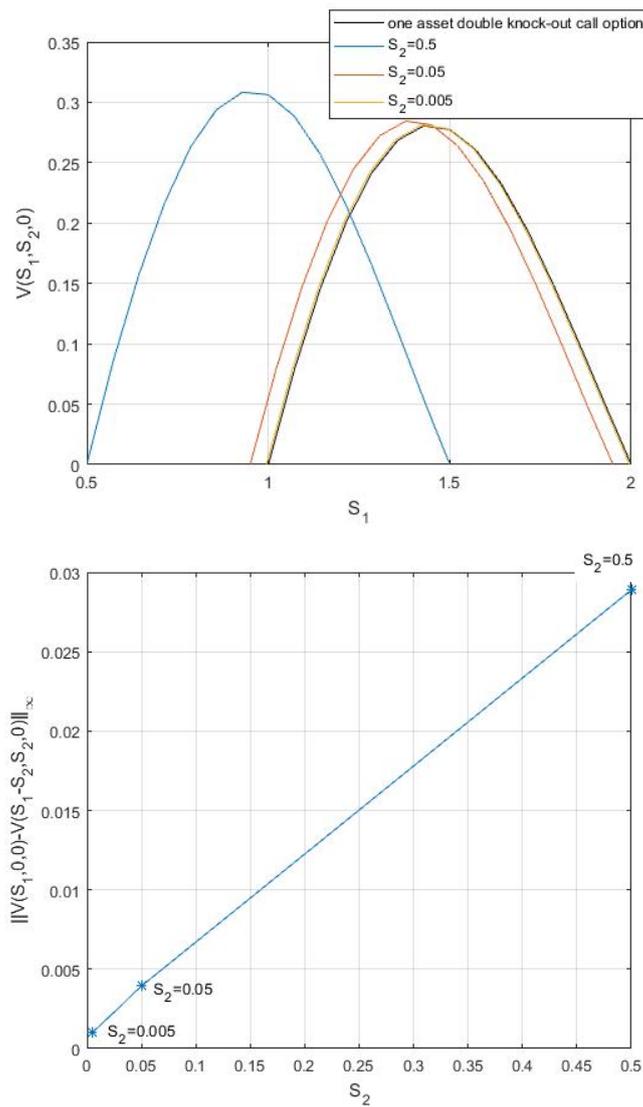
$$\begin{aligned}
 C(S, t) &= S \sum_{n=-\infty}^{\infty} \left\{ \left( \frac{B_2^n}{B_1^n} \right)^{\frac{2r}{\sigma^2}+1} (\mathcal{N}[d_1] - \mathcal{N}[d_2]) - \left( \frac{B_1^{n+1}}{B_2^n S} \right)^{\frac{2r}{\sigma^2}+1} (\mathcal{N}[d_3] - \mathcal{N}[d_4]) \right\} \\
 &\quad - Ke^{-r(T-t)} \sum_{n=-\infty}^{\infty} \left\{ \left( \frac{B_2^n}{B_1^n} \right)^{\frac{2r}{\sigma^2}-1} (\mathcal{N}[d_1 - \sigma\sqrt{T-t}] - \mathcal{N}[d_2 - \sigma\sqrt{T-t}]) \right. \\
 &\quad \left. - \left( \frac{B_1^{n+1}}{B_2^n S} \right)^{\frac{2r}{\sigma^2}-1} (\mathcal{N}[d_3 - \sigma\sqrt{T-t}] - \mathcal{N}[d_4 - \sigma\sqrt{T-t}]) \right\} \tag{29}
 \end{aligned}$$

$$\begin{aligned}
 d_1 &= \frac{\ln\left(\frac{SB_2^{2n}}{KB_1^{2n}}\right) + \left(r + \frac{\sigma_2}{2}(T-t)\right)}{\sigma\sqrt{T-t}} & d_2 &= \frac{\ln\left(\frac{SB_2^{2n-1}}{B_1^{2n}}\right) + \left(r + \frac{\sigma_2}{2}(T-t)\right)}{\sigma\sqrt{T-t}} \\
 d_3 &= \frac{\ln\left(\frac{B_1^{2n+2}}{KS B_2^{2n}}\right) + \left(r + \frac{\sigma_2}{2}(T-t)\right)}{\sigma\sqrt{T-t}} & d_4 &= \frac{\ln\left(\frac{B_1^{2n+2}}{SB_2^{2n+1}}\right) + \left(r + \frac{\sigma_2}{2}(T-t)\right)}{\sigma\sqrt{T-t}}
 \end{aligned}$$

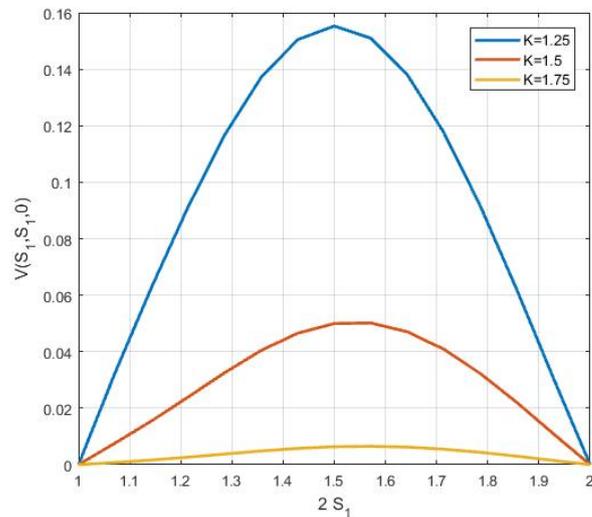
with  $\mathcal{N}$  the standard normal cumulative distribution function.

These “boundary conditions” on  $\Gamma_2$  and  $\Gamma_4$  are necessary to apply Finite Element Method; on the contrary, they are not set during SABO implementation, but they are naturally matched by the approximated solution.

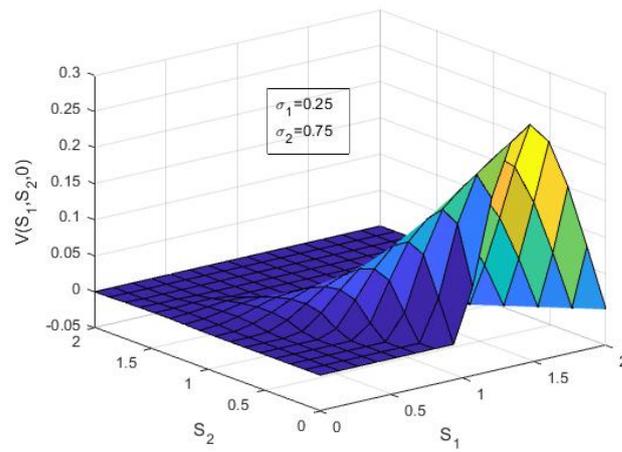
In order to show the stability of results we have changed also some financial parameters starting from Table 1: in Figure 5, the value of the strike price (on the top) and the volatility of one asset (on the bottom) and, in Figure 6, the correlation (poorly correlated assets with  $\rho = 0.1$  on the top, highly correlated assets with  $\rho = 0.9$  on the bottom).



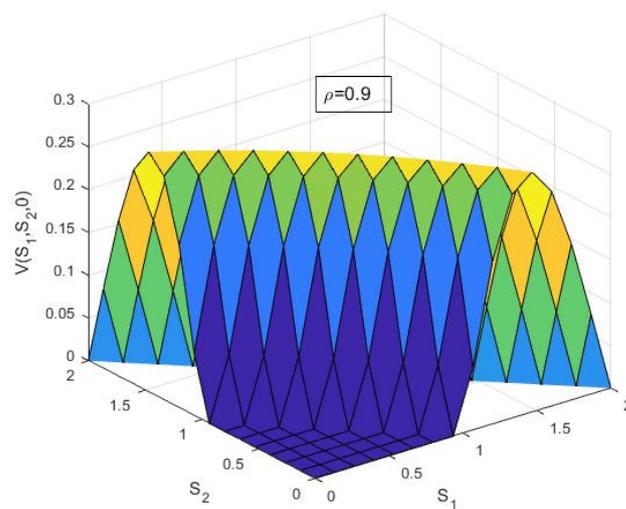
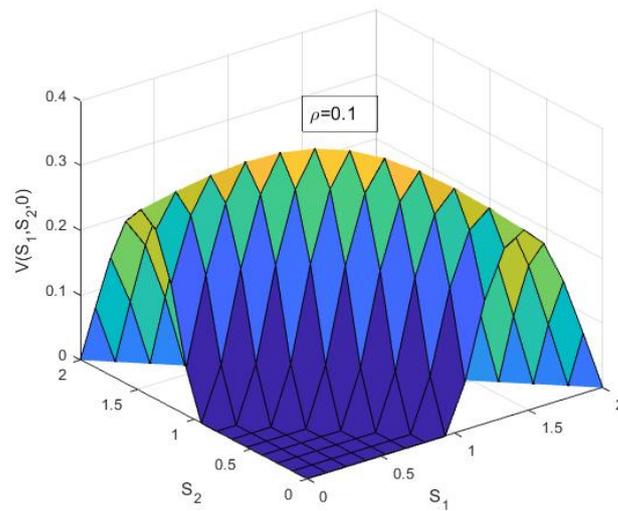
**Figure 4.** Convergence of the solution  $V(S_1, S_2, 0)$  towards  $V(S_1, 0, 0) = C(S_1, 0)$ , exact option value on  $\Gamma_4$ : on the top, projection onto the planes perpendicular to the domain at fixed values of  $S_2$ ; on the bottom, distance to  $C(S_1, 0)$  in  $\infty$ -norm.



**Figure 5.** Cont.



**Figure 5.** Variation of the option value profile  $V(S_1, S_2, 0)$  along the line  $S_1 = S_2$  in relation to the strike  $K$  (top); graph of  $V(S_1, S_2, 0)$  for  $\sigma_1 = 0.25$  and  $\sigma_2 = 0.75$  (bottom).



**Figure 6.** The different aspects of solution for different values of correlation.

- Consider a put down and out option based on two assets whose value is the solution of (1) provided with the payoff function

$$V(S_1, S_2, T) = \max(K - S_2, 0) \quad \text{if } S_1 \in [B, +\infty]$$

meaning that there is a lower out barrier on the first asset only. The boundary and the domain are shown in Figure 7 top.

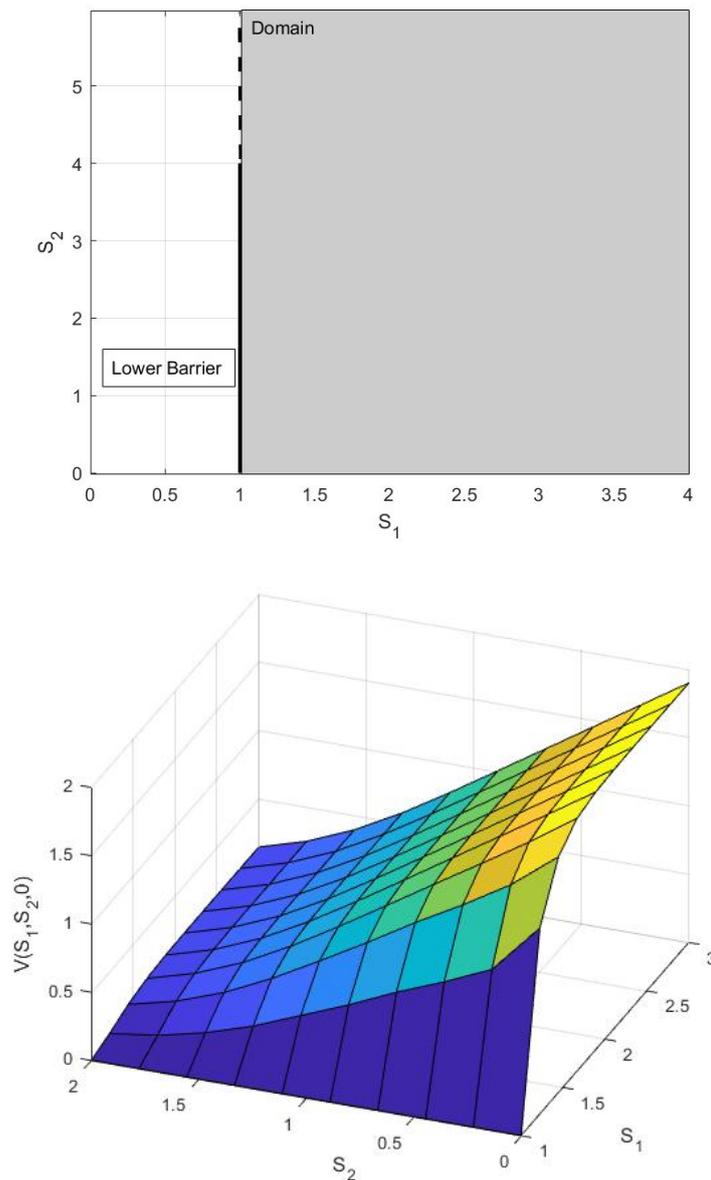


Figure 7. Domain related to second example and solution over a portion of the domain.

In Figure 7 bottom, there is the representation of the approximated solution at time  $t = 0$ , obtained with  $M_{\Delta S} = N_{\Delta t} = 10$ . In Table 3, we can observe the stabilization of digits in the numerical approximation of  $V(S_1, S_2, 0)$  at  $(S_1, S_2) = (2, 1)$  doubling the number of discretization nodes  $M_{\Delta S}$  and  $N_{\Delta t}$  in (22).

**Table 3.** Stabilization of value  $V(S_1, S_2, 0)$  at  $(S_1, S_2) = (2, 1)$  with the refinement of discretization parameters  $\Delta_S$  and  $\Delta_t$ .

$M_{\Delta_S}$ $N_{\Delta_t}$	5	10	20
5	0.89692	0.89731	0.89731
10	0.89679	0.89719	0.89719
20	0.89675	0.89715	0.89716
40	0.89673	0.89714	0.89714

## 7. Conclusions

In this paper, we have extended the SABO method, based on collocation BEM, for pricing barrier options depending on two assets. The good performance of this technique, largely employed in the past for options on a single asset, has been shown by means of numerical results. The implemented Matlab<sup>®</sup> code is enclosed in the Appendix A, in such a way that it can be directly used by any interested reader.

**Author Contributions:** All authors contributed equally to this work. All authors have read and agreed to the published version of the manuscript.

**Funding:** This work has been partially supported by INdAM-GNCS Research Projects.

**Conflicts of Interest:** The authors declare no conflict of interest.

## Appendix A. Matlab<sup>®</sup> Code

All the above provided numerical results were obtained by codes developed with Matlab<sup>®</sup> Release 2007b running on a laptop computer (CPU Intel i5, 4 Gb RAM). The one implementing SABO algorithm applied to the first numerical example in this paper is below available.

```

%Basket options
%Code for the example with data taken from the book of R. Seydel
close all
clear
clc
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
global r sigma1 sigma2 rho
%financial parameters
B2=2; %50; %upper barrier
B1=1; %0.1; %lower barrier
T=1; %expiry
r=0.05; %interest rate
sigma1=0.25; %volatility of S1
sigma2=0.25; %volatility of S2
rho=0.7; %correlation
K=1; %1.5; %strike price
if K>=B2
    disp('strike > upper barrier -> option value = 0')
    return
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%discretization in space
tol=10^-12;
epsi=10^-10;
L1=B2*sqrt(2); %length of Gamma1
L3=B1*sqrt(2); %length of Gamma3
MS1=10; %n. of segments on Gamma1
dS1=B2/MS1;
MS3=ceil(B1/dS1); %n. of segments on Gamma1
dS3=B1/MS3;
L=[L1/MS1:L1 L3/MS3:L3]; %length of each segment
B=[B2*ones(1,MS1) B1*ones(1,MS3)]; %belonged barrier

```



```

%Post processing at time t=0
disp('Post-processing: option value')
NXS1=25;
NXS2=25;
XS1=linspace(0,B2,NXS1);
XS2=linspace(0,B2,NXS2);
[X,Y]=meshgrid(XS1,XS2);
Value=zeros(NXS2,NXS1);
%Values of single asset double knock-out call on Gamma4
for j=1:NXS2
    Value(j,1)=doubleOUT_call(XS2(j),K,B1,B2,T,r,sigma2^2);
end
%Values of single asset double knock-out call on Gamma2
for i=1:NXS1
    Value(1,i)=doubleOUT_call(XS1(i),K,B1,B2,T,r,sigma1^2);
end
for i=2:NXS1
    for j=2:NXS2
        Value(j,i)=0;
        if (XS1(i)+XS2(j)>B1&&XS1(i)+XS2(j)<B2)
            Value(j,i)=integral2(@(S1,S2) (S1+S2-K).*...
                Solfond(XS1(i),XS2(j),0,S1,S2,T),...
                0+epsi,MAX-epsi,Smin,Smax);
            Value(j,i)=Value(j,i)+...
                integral2(@(S1,S2) (S1+S2-K).*...
                    Solfond(XS1(i),XS2(j),0,S1,S2,T),...
                    MAX,B2,0+epsi,Smax);
            for ell=1:1:Nt
                for m=1:MS1+MS3
                    SUM(m,ell)=Alpha(m,ell)*integral2(@(tau,l)...
                        Solfond(XS1(i),XS2(j),0,l,B(m)-l,tau),...
                        t(ell),t(ell+1),x0(m),x1(m));
                end
            end
            Value(j,i)=Value(j,i)+sum(sum(SUM,1),2);
        end
    end
end
end
surf(X,Y,Value)
xlabel('S1')
ylabel('S2')

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function G=Solfond(SS1,SS2,tt,S1,S2,t)
% fundamental solution
global r sigma1 sigma2 rho
alpha1=(log(SS1./S1)+(r-sigma1^2/2)*(t-tt))./(sigma1*sqrt(t-tt));
alpha2=(log(SS2./S2)+(r-sigma2^2/2)*(t-tt))./(sigma2*sqrt(t-tt));
G=exp(-r*(t-tt))./(2*pi*(t-tt))./sqrt(1-rho^2)/(sigma1*sigma2)./...
    (S1.*S2).*exp(-0.5*(alpha1.^2+alpha2.^2-2*alpha1.*alpha2*rho)/...
    (1-rho^2));
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function c=doubleOUT_call(S,K,L,U,t,r,sigma2)
% doubleOUT_call evaluates the call option with double out barrier
% according to the formula in Section 3 in "The complete guide to
% option pricing formulas by E. G. Haug
%
% Inputs:
% S: underlying asset value
% K: strike price
% L: lower barrier
% U: upper barrier
% t: time to maturity (0 not allowed!!!!)
% r: riskless interest rate
% sigma2: variance of the assrt price
% Output:
% c: value of the option at S at time t

```

```

%
if S<=L || S>=U
    c=0;
else
    N=5; % The sum would be infinite , this is the approximation bound

    % Compute some parameters that do not change in the sum. it is therefore
    % convenient to compute them once for all
    sigma = sqrt(sigma2);
    m = (2*r)/sigma2 + 1;
    A = (r+0.5*sigma2)*t;
    B = sigma*sqrt(t);

    sum1=0;
    sum2=0;
    for ndx=-N:N
        % Define parameters that change with ndx
        d1 = (log(S*U^(2*ndx)/(K*L^(2*ndx))) + A)/B;
        d2 = (log(S*U^(2*ndx-1)/(L^(2*ndx))) + A)/B;
        d3 = (log((L^(2*ndx+2))/(K*S*U^(2*ndx))) + A)/B;
        d4 = (log((L^(2*ndx+2))/(S*U^(2*ndx+1))) + A)/B;

        % Update the sums value
        sum1 = sum1 + (U/L)^(ndx*m) * (normcdf(d1)-normcdf(d2)) - ...
            ((L^(ndx+1))/(S*U^ndx)).^m .* (normcdf(d3)-normcdf(d4));
        sum2 = sum2 + (U/L)^(ndx*(m-2)) .* (normcdf(d1-B)-normcdf(d2-B)) - ...
            ((L^(ndx+1))/(U^ndx*S)).^(m-2).* (normcdf(d3-B)-normcdf(d4-B));
    end

    c = S .* sum1 - K*exp(-r*t) .* sum2;
end
end
end

```

## References

- Black, F.; Scholes, M. The Pricing of Options and Corporate Liabilities. *J. Political Econ.* **1973**, *81*, 637–654. [\[CrossRef\]](#)
- Guardasoni, C. Semi-Analytical method for the pricing of barrier options in case of time-dependent parameters (with Matlab® codes). *Commun. Appl. Ind. Math.* **2018**, *9*, 42–67. [\[CrossRef\]](#)
- Guardasoni, C.; Sanfelici, S. Fast numerical pricing of barrier options under stochastic volatility and jumps. *SIAM J. Appl. Math.* **2016**, *76*, 27–57. [\[CrossRef\]](#)
- Aimi, A.; Diazzzi, L.; Guardasoni, C. Efficient BEM-based algorithm for pricing floating strike Asian barrier options (with MATLAB® code). *Axioms* **2018**, *7*, 40. [\[CrossRef\]](#)
- Aimi, A.; Guardasoni, C. Collocation Boundary Element Method for the pricing of Geometric Asian Options. *Eng. Anal. Bound. Elem.* **2018**, *92*, 90–100. [\[CrossRef\]](#)
- Guillaume, T. On the multidimensional Black Scholes partial differential equation. *Ann. Oper. Res.* **2019**, *281*, 229–251. [\[CrossRef\]](#)
- Glasserman, P.; Staum, J. Conditioning on one-step survival for barrier options simulations. *Oper. Res.* **2001**, *49*, 923–937. [\[CrossRef\]](#)
- Cuomo, S.; Di Lorenzo, E.; Di Somma, V.; Toraldo, G. A sequential Monte Carlo approach for the pricing of barrier option under a stochastic volatility model. *Electron. J. Appl. Stat. Anal.* **2020**, *13*, 128–145.
- Giles, M.B. Multilevel Monte Carlo Path Simulation. *Oper. Res.* **2008**, *56*, 607–617. [\[CrossRef\]](#)
- Lars Kirkby, J.; Nguyen, D.; Nguyen, D. A general continuous time Markov chain approximation for multi-asset option pricing with systems of correlated diffusion. *Appl. Math. Comput.* **2020**, *386*, 125472.
- Leentvaar, C.; Oosterlee, C. Multi-asset option pricing using a parallel Fourier-based technique. *J. Comput. Financ.* **2008**, *12*, 1–26. [\[CrossRef\]](#)
- Seydel, R.U. *Tools for Computational Finance*; Springer: Berlin, Germany, 2009.
- Friedman, A. *Partial Differential Equations of Parabolic Type*; Prentice-Hall, Inc.: Englewood Cliffs, NJ, USA, 1964.
- Wilmott, P. *Derivatives: The Theory and Practice of Financial Engineering*; John Wiley: Hoboken, NJ, USA, 1998.
- Haug, E. *The Complete Guide to Option Pricing Formulas*; McGraw-Hill: New York, NY, USA, 2007.
- Ballestra, L.; Pacelli, G. A boundary element method to price time-dependent double barrier options. *Appl. Math. Comput.* **2011**, *218*, 4192–4210. [\[CrossRef\]](#)
- Carr, P.; Itkin, A.; Muravey, D. Semi-closed form prices of barrier options in the time-dependent CEV and CIR models. *J. Deriv.* **2020**, *28*, 26–50. [\[CrossRef\]](#)
- Escobar, M.; Ferrando, S. Barrier options in three dimensions. *Int. J. Financ. Mark. Deriv.* **2014**, *3*, 260–292. [\[CrossRef\]](#)

- 
19. Todorov, V.; Dimov, I. Monte Carlo methods for multidimensional integration for European option pricing. *AIP Conf. Proc.* **2016**, *1773*, 100009. [[CrossRef](#)]
  20. Aimi, A.; Diazzi, L.; Guardasoni, C. Numerical pricing of geometric asian options with barriers. *Math. Methods Appl. Sci.* **2018**, *41*, 7510–7529. [[CrossRef](#)]
  21. Kunimoto, N.; Ikeda, M. Pricing options with curved boundaries. *Math. Financ.* **1992**, *2*, 275–298. [[CrossRef](#)]