NUMERICAL PRICING OF GEOMETRIC ASIAN OPTIONS WITH BARRIERS

A. AIMI*, L. DIAZZI**, C. GUARDASONI*
*DEPARTMENT OF MATHEMATICAL, PHYSICAL AND COMPUTER SCIENCES
UNIVERSITY OF PARMA (ITALY)
MEMBERS OF THE INDAM-GNCS RESEARCH GROUP
**DEPARTMENT OF PHYSICS, INFORMATICS AND MATHEMATICS
UNIVERSITY OF MODENA AND REGGIO EMILIA (ITALY)

Abstract. In this article, a Semi-Analytical method for pricing of Barrier Options (SABO) is described and applied in the context of Asian options with geometric mean. The efficiency of the method is tested and compared with two finite difference methods.

Key words. Boundary Element Method, Finite Difference Method, Geometric Asian Options, Barrier Options.

1. Introduction. The variety of financial products in recent years grew much quicker than ever before. The recent financial crisis has highlighted the need for a more scientific approach to the problem of pricing of these products, taking advantage of more advanced statistical and mathematical skills and of the availability of numerical techniques and faster computer systems. In particular in this paper we will illustrate a new algorithm, the so-called SABO (Semi- Analytical method for pricing of Barrier Options), to evaluate Asian options with barriers. Asian options are derivative contracts whose payoff at maturity depends on the (geometric or arithmetic) average value of an underlying asset over some time interval; in the case of "barrier option", these contracts get into existence or extinguish when the underlying asset reaches a certain barrier value. With respect to European vanilla option, the buyer has a reasonable protection against inconvenient fluctuations in the underlying price and the issuer can attain a better forecasting of the terminal position. For standard Asian options with geometric mean equipped with floating or fixed strike price, closed formula solutions are available for example in Ref. [14] but if the contract involves non standard payoffs or arithmetic mean or barriers, numerical techniques are unavoidable. The pricing is then traditionally based on Monte Carlo methods or on domain methods, such as Finite Volumes and Finite Differences, but Monte Carlo methods are affected by high computational costs and inaccuracy due to their slow convergence and domain methods have some troubles particularly in unbounded domains due to the need of applying artificial and therefore sometimes inaccurate boundary conditions. For a deep survey on the matter, look at Ref. [6]. Barrier options are largely exchanged as they are good products for hedging and investment and they are cheaper than vanilla options but, for Asian options, we found in literature only the analysis of Ref. [2] which provides rigorous bounds in the arithmetic mean case.

SABO has been recently introduced for the computation of European barrier options within various differential models in Refs. [7, 8, 9]. This new approach, based on Boundary Element Method Refs. [3, 4], turns out to be stable and efficient especially when the differential problem is defined in an unbounded domain and the data are assigned on a limited boundary (which is the case of the "barrier options"). The method is particularly advantageous for its high accuracy, for the implicit satisfaction of the far-field behavior of the solution and for the low discretization costs. Moreover it provides a straight hedging computation. The essential requisite, that makes it not

as general as other numerical methods is that, for its application, we need the knowledge of the transition probability density related to the vanilla option problem, at least in an approximated form. The method is here applied to Asian barrier options evaluated with geometric mean that, although not common among practitioners, give some information also about the evaluation of Asian barrier options with arithmetic mean. The obtained results are then compared with numerical results obtained by two finite difference methods among the great variety that we can find in literature (resumed for example in Ref. [11]). The application of finite difference methods is not straightforward since explicit boundary conditions have to be derived from financial conditions in such a way to ensure wellposedness of the starting differential problem Ref. [12] and to avoid oscillations.
Anyway finite difference methods result to be very slow and computing power and time are necessary to reduce the error (as concluded also in Ref. [11] in the case of arithmetic Asian options[1]), therefore SABO seems to be a numerical method that overcomes this issue. The method is developed in the paper for continuously sampled geometric Asian option with no restriction w.r.t. payoff functions. From a theoretical point of view the method is extensible to arithmetic Asian option too with slight modification but, from numerical point of view, there are several troubles that we are going to investigate in the next future. The paper is structured as follows: Sec. 2 introduces the model problem, while SABO is described in Sec. 3. Sec. 4 presents finite difference schemes used for comparison and numerical results are reported in Sec. 5. Conclusions are briefly drawn in Sec. 6.

2. The differential model problem. In the geometric Asian option contract the final value V of the option depends not only on the stock price S_t but also on its evolution through the duration of the contract (assumed to be $[0, T]$) and specifically on its average. The determination of the average value of the underlying prices depends on the following factors: the time detection interval; the type of average that can be arithmetic or geometric; the weight assigned to each price depending on the importance of the period. In this paper we are going to consider the geometric average of the stock price over $[0, T]$. Defining the stochastic process[2]

$$A_t := \int_0^t \log(S_\tau) d\tau \tag{2.1}$$

[1] "...the differential operator... has degenerated elliptic S and A parts... Such degeneracy could be expected to generate oscillations in numerical solutions which might be avoided by adding some (non negative) amount of artificial viscosity... Adding artificial viscosity generally smoothens out a numerical solution, but also decreases the precision since we are now considering a different problem". In the paper, J.E. Zhang investigates a lot of finite difference methods concluding that "The first order consistent time stepping is showing linear convergence in the τ variable. Instead both the first and second order consistent methods in the S variable are showing only linear convergence in this variable. Finally, the linearly consistent method in the A variable is showing a very slow 1/2 order convergence in this variable".

[2] When A and S are written with subscripts (A_t and S_t) they are intended as stochastic processes, otherwise, they are considered independent variables in the differential analysis context.

then the geometric average is $\exp\left(\frac{A_t}{t}\right)$ and the payoff may define different types of Asian option contract, among which:

$$\text{floating strike call} V(S,A,T) = \max\left(S - \exp\left(\frac{A}{T}\right), 0\right) \tag{2.2}$$

$$\text{floating strike put} V(S,A,T) = \max\left(\exp\left(\frac{A}{T}\right) - S, 0\right) \tag{2.3}$$

$$\text{fixed strike call} V(S,A,T) = \max\left(\exp\left(\frac{A}{T}\right) - E, 0\right) \tag{2.4}$$

$$\text{fixed strike put} V(S,A,T) = \max\left(E - \exp\left(\frac{A}{T}\right), 0\right) \tag{2.5}$$

for $S \in \mathbb{R}^+$, $A \in \mathbb{R}$ and E the strike price.
By consequence the Asian option value depends on (S, A, t) through its whole duration and it is known (for example from Refs. [16, 17]) that, if S_t is modeled by a geometric Brownian motion, it satisfies the partial differential equation (PDE):

$$\frac{\partial V}{\partial t} + \frac{\sigma^2}{2} S^2 \frac{\partial^2 V}{\partial S^2} + rS\frac{\partial V}{\partial S} + \log(S)\frac{\partial V}{\partial A} - rV = 0 \qquad S \in \mathbb{R}^+,\ A \in \mathbb{R},\ t \in [0, T) \tag{2.6}$$

where r denotes the risk free interest rate and σ the volatility.
The existence and uniqueness for (2.6) are stated in Ref. [15] and the exact option value $V(S, A, t)$ is known to be the payoff expected value:

$$V(S,A,t) = \int_{-\infty}^{+\infty}\int_{0}^{+\infty} V(\widetilde{S}, \widetilde{A}, T) G(S, A, t; \widetilde{S}, \widetilde{A}, T) d\widetilde{S}\, d\widetilde{A}\,, \tag{2.7}$$

provided that the *transition probability density function* $G(S, A, t; \widetilde{S}, \widetilde{A}, \widetilde{t})$ is available.
From Refs. [5, 1] we have[3]:
(2.8)

$$G(S,A,t;\widetilde{S},\widetilde{A},\widetilde{t}) = \frac{\sqrt{3}H[\widetilde{t}-t]}{\pi\sigma^2(\widetilde{t}-t)^2} \exp\left\{ -\frac{2}{\sigma^2(\widetilde{t}-t)}\log^2\left(\frac{S}{\widetilde{S}}\right) + \frac{6}{\sigma^2(\widetilde{t}-t)^2}\log\left(\frac{S}{\widetilde{S}}\right)\left(A - \widetilde{A} + (\widetilde{t}-t)\log(S)\right) \right.$$
$$\left. -\frac{6}{\sigma^2(\widetilde{t}-t)^3}\left(A - \widetilde{A} + (\widetilde{t}-t)\log(S)\right)^2 - \left(\frac{2r+\sigma^2}{2\sqrt{2}\sigma}\right)^2(\widetilde{t}-t)\right\}\left(\frac{\widetilde{S}}{S}\right)^{\frac{2r-\sigma^2}{2\sigma^2}} \frac{1}{\widetilde{S}}\,.$$

When considering the particular payoff related to fixed strike options or floating strike options, the exact solution can be evaluated also by more efficient closed-formulas of Black-Scholes type that involve normal cumulative distributions (Refs. [14, 18]). In this starting model problem some barriers can be inserted, as often done with European options, but in this case no closed-formulas are available; as example of a barrier contract, a geometric Asian up-and-out barrier call option is an option that is extinguished when the price of the underlying asset grows up enough to breach an assigned upper barrier B before the expiry date T. Barrier options are very common among practitioners because they are cheaper than vanilla option and because they give greater protection against excessive fluctuations of strike price.
The modeling differential problem for a geometric Asian option with up-and-out barrier is:

$$\frac{\partial V}{\partial t} + \frac{\sigma^2}{2} S^2 \frac{\partial^2 V}{\partial S^2} + rS\frac{\partial V}{\partial S} + \log(S)\frac{\partial V}{\partial A} - rV = 0\,, \quad S \in (0, B),\ A \in \mathbb{R},\ t \in [0, T) \tag{2.9}$$

[3] $H[\cdot]$ denotes the Heaviside step function.

$$V(S,A,T) \quad \text{assigned}\,, \quad S\in(0,B),\ A\in\mathbb{R} \tag{2.10}$$

$$V(B,A,t)=0\,, \quad A\in\mathbb{R},\ t\in[0,T) \tag{2.11}$$

(2.12)
$$\text{asymptotic conditions of vanilla option}\,, \quad \{(S,A): S=0\vee A\to-\infty\vee A\to+\infty\}.$$

The method, that we will illustrate in the following section for the solution of (2.9)-(2.12), is rather flexible; therefore it can be easily extended also to Asian call options with other types of barrier, that widen or contract (moving barriers), and to put options, too.
Note that boundary conditions (2.12) are not explicitly assigned. Explicit boundary conditions are not available in literature but some boundary conditions are implicitly embodied in the representation formula (2.7) and they are such to assure existence and uniqueness of the Cauchy partial differential problem solution (look at Ref.[1]).

3. Semi-Analytical method for Barrier Options pricing. SABO is the acronym of **S**emi-**A**nalytical method for the pricing of **B**arrier **O**ptions and it has been introduced for European options both in Black-Scholes Refs. [7, 8] and in Heston and Bates Ref. [9] frameworks. For geometric Asian option a formal and deep argumentation is described in Ref. [1] and it proceeds as follows.

- **The integral representation formula in the domain of the differential problem**

The value $V(S,A,t)$ of an up-and-out barrier call option satisfying the differential problem (2.9)-(2.12) is given by the integral representation formula proven in Ref. [1]:

$$\begin{aligned} V(S,A,t) = &\int_{-\infty}^{+\infty}\int_0^B V(\widetilde{S},\widetilde{A},T)G(S,A,t;\widetilde{S},\widetilde{A},T)d\widetilde{S}\,d\widetilde{A} \\ &\int_t^T\int_{-\infty}^{+\infty}\frac{\sigma^2}{2}B^2\frac{\partial V}{\partial\widetilde{S}}(B,\widetilde{A},\widetilde{t})G(S,A,t;B,\widetilde{A},\widetilde{t})d\widetilde{A}\,d\widetilde{t}\,. \end{aligned} \tag{3.1}$$

at every point (S,A,t) in the existence domain $\Omega\times[0,T)$ with $\Omega:=(0,B)\times\mathbb{R}$.

- **The boundary integral equation (BIE)**

In the integral formula (3.1) $\frac{\partial V}{\partial\widetilde{S}}(B,\widetilde{A},\widetilde{t})$ is unknown so we cannot apply directly (3.1) to get the solution over the whole domain $\Omega\times[0,T)$. But if we consider the limit for $S\to B$ in (3.1) and apply the boundary condition (2.11), we obtain the BIE

$$\begin{aligned} 0=V(B,A,t) = &\int_{-\infty}^{+\infty}\int_0^B V(\widetilde{S},\widetilde{A},T)G(B,A,t;\widetilde{S},\widetilde{A},T)d\widetilde{S}\,d\widetilde{A} \\ &+\int_t^T\int_{-\infty}^{+\infty}\frac{\sigma^2}{2}B^2\frac{\partial V}{\partial\widetilde{S}}(B,\widetilde{A},\widetilde{t})G(B,A,t;B,\widetilde{A},\widetilde{t})d\widetilde{A}\,d\widetilde{t}\,. \end{aligned} \tag{3.2}$$

in the sole unknown $\frac{\partial V}{\partial \widetilde{S}}(B, \widetilde{A}, \widetilde{t})$.
Next goal is to implement a strategy for numerically solving (3.2). The approximation of $\frac{\partial V}{\partial \widetilde{S}}(B, \widetilde{A}, \widetilde{t})$ will be then inserted in the representation formula (3.1), to get the solution V at every desired point of the domain $\Omega \times [0, T)$.

- **Collocation method for the numerical resolution of** (3.2)

We are going to approximate the BIE unknown $\frac{\partial V}{\partial \widetilde{S}}(B, \widetilde{A}, \widetilde{t})$ by collocation method as done in Ref. [1].
The time interval $[0, T]$ is uniformly decomposed by means of

$$\Delta t := \frac{T}{N_t}, \quad N_t \in \mathbb{N}^+, \qquad t_k := k\Delta t, \quad k = 0, \ldots, N_t \tag{3.3}$$

and the BIE unknown is represented in time by piecewise constant basis functions

$$\varphi_k(\widetilde{t}) := H[\widetilde{t} - t_{k-1}] - H[\widetilde{t} - t_k], \quad k = 1, \ldots, N_t\,.$$

The unbounded A-domain $\equiv \mathbb{R}$ needs[4] to be truncated by $[A_{\min}, A_{\max}]$. This cut is substantially involved only in the evaluation of the second term of (3.2) whose integrand function is dominated by the behavior of the transition probability density G. A suitable choice of bounds can be determined by the following considerations and looking at the graph of Erf function[5] (Fig. 1):

$$\int_t^T \int_{-\infty}^{+\infty} G(B, A, t; B, \widetilde{A}, \widetilde{t}) d\widetilde{A}\, d\widetilde{t}$$

$$= -\int_t^T \frac{\exp\left(-\frac{(\widetilde{t}-t)(2r+\sigma^2)^2}{8\sigma^2}\right)}{2B\sigma\sqrt{2\pi(\widetilde{t}-t)}} \mathrm{Erf}\left[\frac{\sqrt{6}\big(A - \widetilde{A} + (\widetilde{t}-t)\log(B)\big)}{(\widetilde{t}-t)^{3/2}\sigma}\right]\Bigg|_{\widetilde{A}\to-\infty}^{\widetilde{A}\to+\infty} d\widetilde{t}$$

$$= \int_t^T \frac{\exp\left(-\frac{(\widetilde{t}-t)(2r+\sigma^2)^2}{8\sigma^2}\right)}{B\sigma\sqrt{2\pi(\widetilde{t}-t)}}\, d\widetilde{t} = \frac{2\,\mathrm{Erf}\left[\frac{(2r+\sigma^2)\sqrt{T-t}}{2\sigma\sqrt{2}}\right]}{B(2r+\sigma^2)}$$

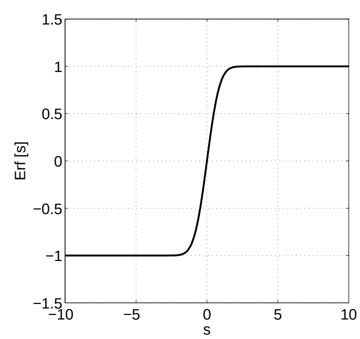

Fig. 1: Graph of Erf function.

hence, we can properly define $A_{\max}$ as the root of the non linear equation below

$$-\int_t^T \frac{\exp\left(-\frac{(\widetilde{t}-t)(2r+\sigma^2)^2}{8\sigma^2}\right)}{2B\sigma\sqrt{2\pi(\widetilde{t}-t)}} \mathrm{Erf}\left[\frac{\sqrt{6}\big(A - A_{\max} + (\widetilde{t}-t)\log(B)\big)}{(\widetilde{t}-t)^{3/2}\sigma}\right] d\widetilde{t} = \frac{\mathrm{Erf}\left[\frac{(2r+\sigma^2)\sqrt{T-t}}{2\sigma\sqrt{2}}\right]}{B(2r+\sigma^2)} \tag{3.4}$$

[4] Thought, in Ref.[1], we have explored the possibility to avoid truncation and using instead infinite elements with equivalent numerical results.

[5] The definition of the *Error Function* is

$$\mathrm{Erf}[z] := \frac{2}{\sqrt{\pi}} \int_0^z e^{-s^2} ds$$

and that of its complement is $\mathrm{Erfc}[z] := 1 - \mathrm{Erf}[z]$.

and $A_{\min}$ as the root of the following non linear equation below

(3.5)
$$\int_t^T \frac{\exp\left(-\frac{(\widetilde{t}-t)(2r+\sigma^2)^2}{8\sigma^2}\right)}{2B\sigma\sqrt{2\pi(\widetilde{t}-t)}} \mathrm{Erf}\left[\frac{\sqrt{6}\left(A - A_{\min} + (\widetilde{t}-t)\log(B)\right)}{(\widetilde{t}-t)^{3/2}\sigma}\right] d\widetilde{t} = \frac{\mathrm{Erf}\left[\frac{(2r+\sigma^2)\sqrt{T-t}}{2\sigma\sqrt{2}}\right]}{B(2r+\sigma^2)}$$

approximating them by Matlab *fzero* function with a tolerance equal to 10^{-10}. Afterwards, we state

$$\Delta A := \frac{A_{\max} - A_{\min}}{N_A}, \quad N_A \in \mathbb{N}^+, \qquad A_h := A_{\min} + h\Delta A, \quad h = 0, \ldots, N_A \tag{3.6}$$

and the representation of the BIE unknown in the independent variable A is given by piecewise constant basis functions as

$$\psi_h(\widetilde{A}) := H[\widetilde{A} - A_{h-1}] - H[\widetilde{A} - A_h], \quad h = 1, \ldots, N_A\,.$$

In Ref. [1], ψ_0 and ψ_{N_A} are chosen as infinite elements over $(-\infty, A_1)$ and $(A_{N_A-1}, +\infty)$ respectively but the choice of all elements equal on $[A_{\min}, A_{\max}]$ gives us a little computational saving and the difference in numerical results is insignificant because, outside the interval $[A_{\min}, A_{\max}]$, the integrals with kernel G involved in (3.1) are negligible. The BIE unknown is then approximated by

$$\frac{\partial V}{\partial \widetilde{S}}(B, \widetilde{A}, \widetilde{t}) \approx \sum_{k=1}^{N_t} \sum_{h=1}^{N_A} \alpha_h^{(k)} \psi_h(\widetilde{A}) \varphi_k(\widetilde{t}) \tag{3.7}$$

and the boundary integral equation (3.2) is evaluated at the collocation points $(\overline{A}_i, \overline{t}_j)$ chosen as the centers of intervals $[A_{i-1}, A_i], [t_{j-1}, t_j]$, i.e.

$$\overline{A}_i = \frac{A_i + A_{i-1}}{2}, \quad i = 1, \ldots, N_A \qquad \overline{t}_j = \frac{t_j + t_{j-1}}{2}, \quad j = 1, \ldots, N_t\,.$$

The resulting linear system of $N_A \times N_t$ equations is:

$$\begin{aligned} \int_{\overline{t}_j}^T \int_{-\infty}^{+\infty} \frac{\sigma^2}{2} B^2 \sum_{k=1}^{N_t} \sum_{h=1}^{N_A} \alpha_h^{(k)} \psi_h(\widetilde{A}) \varphi_k(\widetilde{t}) G(B, \overline{A}_i, \overline{t}_j; B, \widetilde{A}, \widetilde{t})\, d\widetilde{A}\, d\widetilde{t} = \\ - \int_{-\infty}^{+\infty} \int_0^B V(\widetilde{S}, \widetilde{A}, T) G(B, \overline{A}_i, \overline{t}_j; \widetilde{S}, \widetilde{A}, T)\, d\widetilde{S}\, d\widetilde{A} \end{aligned}$$

for $i = 1, \ldots, N_A$, $j = 1, \ldots, N_t$

in matrix form

$$\mathcal{A}\alpha = \mathcal{F}\,, \tag{3.8}$$

where the unknowns are the coefficients of linear representation in (3.7)

$$\alpha = (\alpha^{(k)}\big|_{k=1,\ldots,N_t}) = \left((\alpha_h^{(k)}\big|_{h=1,\ldots,N_A})\big|_{k=1,\ldots,N_t}\right).$$

The matrix entries are:

for $i,h=1,\dots,N_A,\ j,k=1,\dots,N_t$

$$\mathcal{A}_{ih}^{(jk)} = \frac{\sigma^2}{2}B^2\int_{\bar t_j}^{T}\int_{-\infty}^{+\infty}\psi_h(\widetilde{A})\varphi_k(\widetilde{t})G(B,\overline{A}_i,\bar t_j;B,\widetilde{A},\widetilde{t})\,d\widetilde{A}\,d\widetilde{t} =$$

$$\frac{\sigma^2}{2}B^2 H[t_k-\bar t_j]\int_{\max(t_{k-1},\bar t_j)}^{t_k}\int_{A_{h-1}}^{A_h}\frac{\sqrt{3}}{\pi\sigma^2(\widetilde{t}-\bar t_j)^2 B}$$

$$\exp\left\{-\frac{6\big(\overline{A}_i-\widetilde{A}+(\widetilde{t}-\bar t_j)\log(B)\big)^2}{\sigma^2(\widetilde{t}-\bar t_j)^3}-\left(\frac{2r+\sigma^2}{2\sqrt{2}\sigma}\right)^2(\widetilde{t}-\bar t_j)\right\}d\widetilde{A}\,d\widetilde{t}$$

and, since equation (2.9) has constant parameters w.r.t. t and A variables, they depend only on the difference between time instants and between grid A-points. So, defining $\xi = i-h$, $\xi=-N_A+1,\dots,N_A-1$ and $\ell=k-j$, $\ell=0,\dots,N_t-1$, performing the change of variable $\widetilde{t}=\Delta t(\tau+k-1)$, we get $\widetilde{t}-\bar t_j=\Delta t(\tau+k-j-1/2)=\Delta t(\tau+\ell-1/2)$ and therefore

(3.9)

$$\mathcal{A}_{ih}^{(jk)}=\frac{B\sqrt{3}}{2\pi\Delta t}\int_{\frac12-\frac12 H[\ell]}^{1}\int_{A_{h-1}}^{A_h}\frac{1}{(\tau+\ell-1/2)^2}$$

$$\exp\left\{-\frac{6\big(\overline{A}_i-\widetilde{A}+\Delta t(\tau+\ell-1/2)\log(B)\big)^2}{\sigma^2\Delta t^3(\tau+\ell-1/2)^3}-\left(\frac{2r+\sigma^2}{2\sqrt{2}\sigma}\right)^2\Delta t(\tau+\ell-1/2)\right\}d\widetilde{A}\,d\tau$$

$$=\frac{\sigma B\Delta t}{4\sqrt{2\pi}}\int_{\frac12-\frac12 H[\ell]}^{1}\frac{\exp\left\{-\left(\frac{2r+\sigma^2}{2\sqrt{2}\sigma}\right)^2\Delta t(\tau+\ell-1/2)\right\}}{\sqrt{\Delta t(\tau+\ell-1/2)}}\left\{\mathrm{Erf}\left[\frac{\sqrt{6}\big(\Delta A(\xi+\frac12)+\Delta t(\tau+\ell-1/2)\log(B)\big)}{\sigma\Delta t^{3/2}(\tau+\ell-1/2)^{3/2}}\right]\right.$$

$$\left.-\mathrm{Erf}\left[\frac{\sqrt{6}\big(\Delta A(\xi-\frac12)+\Delta t(\tau+\ell-1/2)\log(B)\big)}{\sigma\Delta t^{3/2}(\tau+\ell-1/2)^{3/2}}\right]\right\}d\tau =: \mathcal{A}_{\xi}^{(\ell)}\,.$$

Interior integration in (3.9) w.r.t. $\widetilde{A}$ cannot be numerically performed if $\ell=0$ because, actually, when $\tau\to1/2$ the exponential function "flattens" the result to zero. Moreover pay attention also to integration w.r.t. τ, due to high gradient regions in the integrand function: we simply performed it by the `quad` Matlab function checking inputs suitability.

We have to observe that $\mathcal{A}$ is therefore a block upper triangular matrix with Toeplitz structure: the upper triangularity is due to the fact that the fundamental solution (2.8) is defined only for $\widetilde{t}>t$ implying that the matrix entries are non trivial only for $k\ge j$; the Toeplitz structure is due to the dependence on time differences. Moreover each block has in turn a Toeplitz structure due to the dependence on grid A-points differences:

(3.10)

$$\mathcal{A}=\begin{bmatrix}\mathcal{A}^{(0)} & \mathcal{A}^{(1)} & \mathcal{A}^{(2)} & \cdots & \mathcal{A}^{(N_t-1)}\\ 0 & \mathcal{A}^{(0)} & \mathcal{A}^{(1)} & \cdots & \mathcal{A}^{(N_t-2)}\\ 0 & 0 & \mathcal{A}^{(0)} & \ddots & \vdots\\ \vdots & \vdots & \ddots & \ddots & \mathcal{A}^{(1)}\\ 0 & 0 & \cdots & 0 & \mathcal{A}^{(0)}\end{bmatrix}\ \text{with}\ \mathcal{A}^{(\ell)}=\begin{bmatrix}\mathcal{A}_0^{(\ell)} & \mathcal{A}_{-1}^{(\ell)} & \mathcal{A}_{-2}^{(\ell)} & \cdots & \mathcal{A}_{-N_A+1}^{(\ell)}\\ \mathcal{A}_1^{(\ell)} & \mathcal{A}_0^{(\ell)} & \mathcal{A}_{-1}^{(\ell)} & \cdots & \mathcal{A}_{-N_A+2}^{(\ell)}\\ \mathcal{A}_2^{(\ell)} & \mathcal{A}_1^{(\ell)} & \mathcal{A}_0^{(\ell)} & \ddots & \vdots\\ \vdots & \vdots & \ddots & \ddots & \mathcal{A}_{-1}^{(\ell)}\\ \mathcal{A}_{N_A-1}^{(\ell)} & \mathcal{A}_{N_A-2}^{(\ell)} & \cdots & \mathcal{A}_1^{(\ell)} & \mathcal{A}_0^{(\ell)}\end{bmatrix},$$

for $\ell=0,..,N_t-1$. This allows to solve the linear system by block backward substitution and to compute only the entries in the first column and in the first row of the

sub-matrices belonging to the last column in the block structure with considerable computational saving.
The rhs entries are:

for $i = 1, \ldots, N_A$, $j = 1, \ldots, N_t$

$$\mathcal{F}_i^{(j)} = -\int_{-\infty}^{+\infty}\int_0^B V(\widetilde{S},\widetilde{A},T)G(B,\overline{A}_i,\overline{t}_j;\widetilde{S},\widetilde{A},T)\,d\widetilde{S}\,d\widetilde{A}\,. \tag{3.11}$$

In the case of fixed strike payoff

$$\mathcal{F}_i^{(j)} = -\int_{-\infty}^{+\infty}\int_0^B \max\left(e^{\frac{\widetilde{A}}{T}} - E, 0\right) G(B,\overline{A}_i,\overline{t}_j;\widetilde{S},\widetilde{A},T)\,d\widetilde{S}\,d\widetilde{A} \tag{3.12}$$

(3.13)

$$= \frac{B^{-\frac{2r-\sigma^2}{2\sigma^2}}}{2\sigma\sqrt{2\pi(T-\overline{t}_j)}}\int_0^B \widetilde{S}^{\frac{2r-3\sigma^2}{2\sigma^2}+\frac{T-\overline{t}_j}{2T}} \exp\left(-\frac{(T-\overline{t}_j)^2(2r+\sigma^2)^2 + 4\log^2\left(\frac{B}{\widetilde{S}}\right)}{8\sigma^2(T-\overline{t}_j)}\right)$$
$$\left\{E\widetilde{S}^{-\frac{T-\overline{t}_j}{2T}}\operatorname{Erfc}\left[\sqrt{\frac{3}{2}}\,\frac{-2\overline{A}_i - 2(T-\overline{t}_j)\log(B) + 2T\log(E) + (T-\overline{t}_j)\log(\frac{B}{\widetilde{S}})}{\sigma(T-\overline{t}_j)^{\frac{3}{2}}}\right] + B^{\frac{T-\overline{t}_j}{2T}}\exp\left(\frac{\overline{A}_i}{T} + \frac{\sigma^2(T-\overline{t}_j)^3}{24\,T^2}\right)\right.$$
$$\left.\left(-2 + \operatorname{Erfc}\left[\frac{12\,T\overline{A}_i + \sigma^2(T-\overline{t}_j)^3 - 6\,T\big(-2(T-\overline{t}_j)\log(B) + 2\,T\log(E) + (T-\overline{t}_j)\log(\frac{B}{\widetilde{S}})\big)}{2\sqrt{6}\,T\sigma(T-\overline{t}_j)^{\frac{3}{2}}}\right]\right)\right\}d\widetilde{S}\,.$$

- **The numerical approximation of option price**

Solving system (3.8) we get α (that determines the approximation of BIE solution) and, introducing (3.7) in the integral representation formula (3.1), we obtain an approximation of $V(S,A,t)$ at any domain point $(S,A,t) \in \Omega \times [0,T)$ by evaluating[6]

$$\begin{aligned} V(S,A,t) \approx &\int_{-\infty}^{+\infty}\int_0^B V(\widetilde{S},\widetilde{A},T)G(S,A,t;\widetilde{S},\widetilde{A},T)\,d\widetilde{S}\,d\widetilde{A} + \\ &\frac{\sigma^2}{2}B^2\sum_{k=\mathrm{floor}[\frac{t}{\Delta t}]+1}^{N_t}\sum_{h=1}^{N_A}\alpha_h^{(k)}\int_{\max(t,t_{k-1})}^{t_k}\int_{A_{h-1}}^{A_h} G(S,A,t;B,\widetilde{A},\widetilde{t})\,d\widetilde{A}\,d\widetilde{t}\,. \end{aligned} \tag{3.14}$$

With respect to domain methods, SABO reaches a remarkable computational saving as it is possible to compute the approximate solution at the points of interest avoiding evaluation at all the discretization mesh points.
In the case of fixed strike payoff, the first integral in (3.14) can be computed as in

[6] floor[$\cdot$]:=function that rounds its argument to the nearest integers towards minus infinity.

(3.12):

(3.15)

$$V(S,A,t) \approx -\frac{S^{-\frac{2r-\sigma^2}{2\sigma^2}}}{2\sigma\sqrt{2\pi(T-t)}} \int_0^B \widetilde{S}^{\frac{2r-3\sigma^2}{2\sigma^2}+\frac{T-t}{2T}} \exp\left(-\frac{(T-t)^2(2r+\sigma^2)^2 + 4\log^2\left(\frac{S}{\widetilde{S}}\right)}{8\sigma^2(T-t)}\right)$$

$$\left\{E\widetilde{S}^{-\frac{T-t}{2T}} \operatorname{Erfc}\left[\sqrt{\frac{3}{2}}\,\frac{-2A-2(T-t)\log(S)+2T\log(E)+(T-t)\log(\frac{S}{\widetilde{S}})}{\sigma(T-t)^{\frac{3}{2}}}\right]\right.$$

$$+S^{\frac{T-t}{2T}} \exp\left(\frac{A}{T}+\frac{\sigma^2(T-t)^3}{24T^2}\right)$$

$$\left.\left(-2+\operatorname{Erfc}\left[\frac{12\,TA+\sigma^2(T-t)^3-6\,T\big(-2(T-t)\log(S)+2\,T\log(E)+(T-t)\log(\frac{S}{\widetilde{S}})\big)}{2\sqrt{6}\,T\sigma(T-t)^{\frac{3}{2}}}\right]\right)\right\} d\widetilde{S}$$

$$+\frac{\sigma B}{4\sqrt{2\pi}}\left(\frac{B}{S}\right)^{\frac{2r-\sigma^2}{2\sigma^2}} \sum_{k=\mathrm{floor}[\frac{t}{\Delta t}]+1}^{N_t} \sum_{h=1}^{N_A} \alpha_h^{(k)} \int_{\max(t,t_{k-1})}^{t_k} \frac{1}{\sqrt{\widetilde{t}-t}} \exp\left(-\frac{(\widetilde{t}-t)^2(2r+\sigma^2)^2+4\log^2\left(\frac{S}{B}\right)}{8(\widetilde{t}-t)\sigma^2}\right)$$

$$\left\{\operatorname{Erf}\left[\sqrt{\frac{3}{2}}\frac{2(A_h-A)-2(\widetilde{t}-t)\log(S)+(\widetilde{t}-t)\log\left(\frac{S}{B}\right)}{(\widetilde{t}-t)^{3/2}\sigma}\right]\right.$$

$$\left.-\operatorname{Erf}\left[\sqrt{\frac{3}{2}}\frac{2(A_{h-1}-A)-2(\widetilde{t}-t)\log(S)+(\widetilde{t}-t)\log\left(\frac{S}{B}\right)}{(\widetilde{t}-t)^{3/2}\sigma}\right]\right\} d\widetilde{t}.$$

- **Hedging**

The computation of Greeks can be deduced by derivation of the representation formula (3.1) and using SABO. For the usual example of $\boldsymbol{\Delta}-Greek$, at $t\in[0,T)$, $S\in(0,B)$

$$\begin{aligned}
&\boldsymbol{\Delta}(S,A,t) := \frac{\partial V}{\partial S}(S,A,t)\\
&= \int_{-\infty}^{+\infty}\int_0^B V(\widetilde{S},\widetilde{A},T)\frac{\partial G}{\partial S}(S,A,t;\widetilde{S},\widetilde{A},T)\,d\widetilde{S}\,d\widetilde{A}+ \qquad (3.16)\\
&\frac{\sigma^2}{2}B^2\int_t^T\int_{-\infty}^{+\infty}\frac{\partial V}{\partial \widetilde{S}}(B,\widetilde{A},\widetilde{t})\frac{\partial G}{\partial S}(S,A,t;B,\widetilde{A},\widetilde{t})\,d\widetilde{A}\,d\widetilde{t}
\end{aligned}$$

with

(3.17)

$$\frac{\partial G}{\partial S}(S,A,t;\widetilde{S},\widetilde{A},\widetilde{t}) = G(S,A,t;\widetilde{S},\widetilde{A},\widetilde{t})\frac{-12(A-\widetilde{A})-(\widetilde{t}-t)^2(2r-\sigma^2)-4(\widetilde{t}-t)\log(\widetilde{S}S^2)}{2\sigma^2S(\widetilde{t}-t)^2}.$$

Once get the coefficients α, the approximation (3.7) of $\frac{\partial V}{\partial \widetilde{S}}(B,\widetilde{A},\widetilde{t})$ inserted in (3.16) straightforwardly gives us $\boldsymbol{\Delta}-Greek$ even without computing the primary unknown V and only at the point (S,A,t) where its derivatives are required. The use of the closed form expression (3.16) for $\boldsymbol{\Delta}-Greek$ allows us to get its approximation with far superior accuracy with respect to other finite difference method as will be evident in next numerical examples. This inference could be analogously extended to the computation of the other Greeks $\boldsymbol{\Gamma}$, $\boldsymbol{\Theta}$, $\boldsymbol{\rho}$ and **Vega**.

4. Finite Difference Methods. We are going to check the efficiency of SABO in comparison with Finite Difference (FD) methods because they are, in general, the most efficient numerical methods among the classical ones that can be applied in pricing financial derivatives but the interested reader can look at Ref. [17] for an overall survey of numerical methods in Finance and at Ref. [6] for a survey dedicated to arithmetic Asian Options (that reasonably can be extended to the geometric case). We have chosen two particular FD Methods in the remarkable amount of possibilities, considering ease and practicability in implementation, but in Ref. [11] there is a deep analysis of a large number of them pointing out a generalized slowness in their performance.
Equation (2.9) is proven to be hypoelliptic (Refs. [5, 15, 10]), property that guarantees smooth solution, but it is strongly degenerate: at $S = 0$ because the logarithmic coefficient of $\frac{\partial V}{\partial A}$ is not defined and in A-variable because the term $\frac{\partial^2 V}{\partial A^2}$ lacks. These degenerations may cause oscillations in FD numerical solutions and the adopted repair strategies in general deteriorate accuracy (Ref. [11]).

4.1. Boundary conditions. Boundary conditions (2.12) are non trivially established or deduced, as they are not directly involved in the stochastic arguments for the determination of solution (2.7). Anyway J. Hugger in Ref. [12], where the wellposedness of the boundary value formulation for arithmetic Asian options with fixed strike price is treated, gives us some good hints.
Aiming to evaluate the option at $S = S^*$, $A = A^*$, $t = 0$ by FD methods, we need to reduce the unbounded (S, A)-space $(0, B] \times \mathbb{R}$ to a bounded set $\Omega^* = [S_{\min}, B] \times [A_{\min}, A_{\max}]$ such that $(S^*, A^*) \in \Omega^*$ and then apply artificial boundary conditions at the edges. Note that the choice of bounds and boundary conditions is very tricky and sometimes we cannot avoid to find them empirically.

- We have chosen $A_{\min}$ less or equal to the minimum between the value suggested in (3.5) and A^*.
  From the financial point of view, it is meaningless to consider a condition at $A = -\infty$ for whichever value of S and t and, by consequence, it is not easy to apply a proper condition at points $(S, A_{\min}, t)$, hence we have decided to use an upwind method: if $\log(S) > 0$ then $\frac{\partial V}{\partial A}$ is approximated by a forward difference so that the boundary condition at $A_{\min}$ becomes useless, otherwise by a backward difference, artificially assuming that $\frac{\partial V}{\partial A}(S, A_{\min}, t) = 0$.
- The upper bound is set at $A_{\max} \geq \max(A^*, T\log(E))$.
  If, at time t, $\exp(A/t) - E > 0$ then, at maturity, $\exp(A/T) - E > 0$ since the average $\exp(A/T)$ is a non-decreasing function of time. For $A \geq T\log(E)$ and without barriers, the put Asian options with fixed strike price (2.5) is null and the knowledge of $V(S, A, t)$ in a closed form can be deduced from the related put-call parity
  (4.1)
  $$\begin{aligned} &C_{\text{fix}}(S, A, t) - P_{\text{fix}}(S, A, t) = e^{-r(T-t)}\left(\mathbb{E}[e^{A/T}] - E\right) \\ &= S^{\frac{T-t}{T}} \exp\left(\frac{A}{T} + \left[\left(r - \frac{\sigma^2}{2}\right)\frac{T-t}{2T} + \frac{\sigma^2}{6}\frac{(T-t)^2}{T^2} - r\right](T-t)\right) - Ee^{-r(T-t)}\,. \end{aligned}$$

  By consequence, if $\log(S) > 0$ then $\frac{\partial V}{\partial A}$ is approximated by a forward difference and, provided that $A_{\max}$ is "far enough" from A^*, the boundary

condition at $A_{\max}$ is set as
(4.2)
$$V(S,A_{\max},t)=S^{\frac{T-t}{T}}\exp\left(\frac{A_{\max}}{T}+\left[\left(r-\frac{\sigma^2}{2}\right)\frac{T-t}{2T}+\frac{\sigma^2}{6}\frac{(T-t)^2}{T^2}-r\right](T-t)\right)-Ee^{-r(T-t)}\,.$$

Note that this boundary condition exploits some theoretical knowledge about call Asian options with fixed strike price, hence, the change of payoff condition needs more a priori analytical calculations in FD than in SABO.
Otherwise, if $\log(S)<0$ and $\frac{\partial V}{\partial A}$ is approximated by a backward difference, the condition at $A_{\max}$ becomes useless.

- Looking at S-space, at first we can consider $S_{\min}=0$ even if the Eq. (2.9) is there degenerate.
At $S=0$ the exact boundary condition is inferred from Eq. (2.9):

$$\lim_{S\to 0}\left[\frac{\partial V}{\partial t}+\frac{\sigma^2}{2}S^2\frac{\partial^2 V}{\partial S^2}+rS\frac{\partial V}{\partial S}+\log(S)\frac{\partial V}{\partial A}-rV\right]=\lim_{S\to 0}\left[\frac{\partial V}{\partial t}-rV\right]=0$$

because if $S_t=0$ at any time then the average asset price $\exp(A_t/t)=$ *constant* and the option value V can be considered independent of stochastic variables S and A. Therefore, solving the ordinary differential equation

$$\lim_{S\to 0}\left[\frac{\partial V}{\partial t}-rV\right]=0$$

we derive the condition

$$\lim_{S\to 0}V(S,A,t)=e^{-r(T-t)}\lim_{S\to 0}V(S,A,T)\,. \tag{4.3}$$

We can even try to optimize the computational domain by reducing the possibly excessive S-space with a lower bound $S_{\min}\neq 0$ thought empirically determined. A proper choice of boundary condition at $S_{\min}$ can be the closed-formula (2.7) for the option without barrier that in fixed strike case reduces to one of those available in Ref. [18] for call and put options respectively:
(4.4)
$$C_{\text{fix}}(S,A,t)=S^*\mathcal{N}\left(\frac{\log\frac{S^*}{E}+(r+\frac{\sigma^{*2}}{2})(T-t)}{\sigma^*\sqrt{T-t}}\right)-Ee^{-r(T-t)}\mathcal{N}\left(\frac{\log\frac{S^*}{E}+(r-\frac{\sigma^{*2}}{2})(T-t)}{\sigma^*\sqrt{T-t}}\right)$$
$$S^*=S^{\frac{T-t}{T}}\exp\left(\frac{A}{T}+(\mu^*-r)(T-t)\right)$$
$$\mu^*=\left(r-q-\frac{\sigma^2}{2}\right)\frac{T-t}{2T}+\frac{\sigma^2}{6}\frac{(T-t)^2}{T^2}\,,\quad \sigma^*=\frac{\sigma}{\sqrt{3}}\frac{T-t}{T}$$
$$P_{\text{fix}}(S,A,t)=Ee^{-r(T-t)}\mathcal{N}\left(-\frac{\log\frac{S^*}{E}+(r-\frac{\sigma^{*2}}{2})(T-t)}{\sigma^*\sqrt{T-t}}\right)-S^*\mathcal{N}\left(-\frac{\log\frac{S^*}{E}+(r+\frac{\sigma^{*2}}{2})(T-t)}{\sigma^*\sqrt{T-t}}\right)\,.$$

4.2. Grid and data. Maintaining the grid defined in (3.3) and (3.6) in (A,t)-domain $[A_{\min},A_{\max}]\times[0,T]$, for the implementation of FD schemes we need to introduce the grid in S-domain $[S_{\min},B]$

$$\Delta S:=\frac{B-S_{\min}}{N_S},\quad N_S\in\mathbb{N}^+,\qquad S_i:=S_{\min}+i\Delta S,\quad i=0,\dots,N_S \tag{4.5}$$

and to define the approximated option value

$$V_{i,h}^k :\approx V(S_i, A_h, t_k)\,.$$

The values of $V_{ih}^{N_t}$ can be found from the final condition:

$$V_{i,h}^{N_t} = V(S_i, A_h, T), \quad i = 0, \cdots, N_S-1,\ h = 0, \cdots, N_A\,.$$

For $i = 0, \cdots, N_S-1$, if $\log(S_i) > 0$ at the boundary $A = A_{\max}$ we use the boundary condition (4.2) hence, for $k = 0, \cdots, N_t - 1$, we can evaluate

$$V_{i,N_A}^k = S_i^{\frac{T-t_k}{T}} e^{\frac{A_{\max}}{T}+\left[\left(r-\frac{\sigma^2}{2}\right)\frac{T-t_k}{2T}+\frac{\sigma^2}{6}\frac{(T-t_k)^2}{T^2}-r\right](T-t_k)} - Ee^{-r(T-t_k)}\,;$$

if $\log(S_i) < 0$ at the boundary $A = A_{\min}$ we use the vanishing Neumann boundary condition and hence, for $k = 0, \cdots, N_t - 1$, we assign

$$V_{i,0}^k = V_{i,1}^k\,.$$

At the boundary $S = S_{\min}$, we apply the condition (4.3) or (4.4); hence we have

$$V_{0,h}^k \quad \text{assigned}, \quad \mathrm{h} = 0, \cdots, \mathrm{N_A} - 1,\ \mathrm{k} = 0, \cdots, \mathrm{N_t} - 1\,.$$

At the boundary $S = B$, we use the boundary condition (2.11); hence

$$V_{N_S,h}^k = 0, \quad h = 0, \cdots, N_A,\ k = 0, \cdots, N_t\,.$$

4.3. First FD scheme. The first scheme we take into account is the very classical FD scheme (that we will denote for brevity by FD1): the derivatives of V in (2.9) are approximated by truncations of Taylor expansions. In particular we use:

- first order backward difference for the time derivative approximation

$$\frac{\partial V}{\partial t}(S_i, A_h, t_k) = \frac{V_{i,h}^k - V_{i,h}^{k-1}}{\Delta t} + O(\Delta t)$$

- second order central difference for the S derivative approximation

$$\frac{\partial V}{\partial S}(S_i, A_h, t_k) = \frac{V_{i+1,h}^k - V_{i-1,h}^k}{2\Delta S} + O(\Delta S^2)$$

- second order central difference for the S second order derivative approximation

$$\frac{\partial^2 V}{\partial S^2}(S_i, A_h, t_k) = \frac{V_{i+1,h}^k - 2V_{i,h}^k + V_{i-1,h}^k}{\Delta S^2} + O(\Delta S^2)$$

- if $\log(S_i) > 0$, first order forward difference for the A derivative approximation[7]

$$\frac{\partial V}{\partial A}(S_i, A_h, t_k) = \frac{V_{i,h+1}^k - V_{i,h}^k}{\Delta A} + O(\Delta A)$$

[7] If $\log(S_i) < 0$, we use first order backward difference for the A derivative approximation

$$\frac{\partial V}{\partial A}(S_i, A_h, t_k) = \frac{V_{i,h}^k - V_{i,h-1}^k}{\Delta A} + O(\Delta A)$$

and formula (4.6) is modified accordingly.

We can now write down the discrete approximation of (2.9) at each point (S_i, A_h, t_k):

for $i = 1, \dots, N_S - 1,\ h = 0, \dots, N_A - 1,\ k = 0, \dots, N_t - 1$

(4.6)
$$\frac{V_{i,h}^k - V_{i,h}^{k-1}}{\Delta t} + \frac{1}{2}\sigma^2 S_i^2 \frac{V_{i+1,h}^k - 2V_{i,h}^k + V_{i-1,h}^k}{\Delta S^2} + rS_i \frac{V_{i+1,h}^k - V_{i-1,h}^k}{2\Delta S} + \log(S_i)\frac{V_{i,h+1}^k - V_{i,h}^k}{\Delta A} - rV_{i,h}^k = 0$$

rearranging the scheme in a compact form

$$V_{i,h}^{k-1} = a_i V_{i,h}^k + b_i V_{i+1,h}^k + c_i V_{i-1,h}^k + d_i V_{i,h+1}^k$$

where

$$a_i = 1 - \Delta t\big(r + \sigma^2 i^2 + \tfrac{\log(S_i)}{\Delta A}\big)\,, \quad b_i = \tfrac{\Delta t}{2}\big(ri + \sigma^2 i^2\big)\,, \quad c_i = \tfrac{\Delta t}{2}\big(-ri + \sigma^2 i^2\big)\,, \quad d_i = \tfrac{\Delta t}{\Delta A}\log(S_i)\,.$$

It is easy to see that the scheme, backward computing the new value $V_{i,h}^{k-1}$, is explicit in time. We want to remark two more things: first, that the weights depends only on i therefore on the S variable and, second, that values $V_{i,0}^k$ having $A_{\min}$ as a coordinate give no contribution to the scheme (in accordance with observation in Sec. 4.1).

4.4. Second FD scheme. The idea of this method (that we will denote for brevity by FD2) is taken from Ref. [13] and it is conceived only for the case $\log(S_i) > 0$. The PDE (2.9) is collocated at the points

$$(S_i, A_{h+\frac{1}{2}}, t_{k+\frac{1}{2}}) := \Big(S_i, A_h + \frac{\Delta A}{2}, t_k + \frac{\Delta t}{2}\Big)\,.$$

Instead of partial derivatives, we use the following approximations, based on suitable Taylor expansions and standard finite difference approximations:

$$\begin{aligned}\frac{\partial V}{\partial t}(S_i, A_{h+\frac{1}{2}}, t_{k+\frac{1}{2}}) &= \frac{1}{2}\Big(\frac{\partial V}{\partial t}(S_i, A_h, t_{k+\frac{1}{2}}) + \frac{\partial V}{\partial t}(S_i, A_{h+1}, t_{k+\frac{1}{2}})\Big) + O(\Delta A^2)\\ &= \frac{V_{i,h}^{k+1} - V_{i,h}^k}{2\Delta t} + \frac{V_{i,h+1}^{k+1} - V_{i,h+1}^k}{2\Delta t} + O(\Delta t^2 + \Delta A^2)\,,\end{aligned}$$

$$\begin{aligned}\frac{\partial V}{\partial S}(S_i, A_{h+\frac{1}{2}}, t_{k+\frac{1}{2}}) &= \frac{1}{2}\Big(\frac{\partial V}{\partial S}(S_i, A_{h+1}, t_{k+1}) + \frac{\partial V}{\partial S}(S_i, A_h, t_k)\Big) + O(\Delta t^2 + \Delta A^2)\\ &= \frac{V_{i+1,h+1}^{k+1} - V_{i-1,h+1}^{k+1}}{4\Delta S} + \frac{V_{i+1,h}^k - V_{i-1,h}^k}{4\Delta S} + O(\Delta t^2 + \Delta A^2 + \Delta S^2)\,,\end{aligned}$$

$$\begin{aligned}\frac{\partial^2 V}{\partial S^2}(S_i, A_{h+\frac{1}{2}}, t_{k+\frac{1}{2}}) &= \frac{1}{2}\Big(\frac{\partial^2 V}{\partial S^2}(S_i, A_{h+1}, t_{k+1}) + \frac{\partial^2 V}{\partial S^2}(S_i, A_h, t_k)\Big) + O(\Delta t^2 + \Delta A^2)\\ &= \frac{V_{i-1,h+1}^{k+1} - 2V_{i,h+1}^{k+1} + V_{i+1,h+1}^{k+1}}{2\Delta S^2} + \frac{V_{i-1,h}^k - 2V_{i,h}^k + V_{i+1,h}^k}{2\Delta S^2} + O(\Delta t^2 + \Delta A^2 + \Delta S^2)\,,\end{aligned}$$

$$V(S_i, A_{h+\frac{1}{2}}, t_{k+\frac{1}{2}}) = \frac{1}{2}\Big(V_{i,h}^k + V_{i,h+1}^{k+1}\Big) + O(\Delta t^2 + \Delta A^2)\,,$$

$$\begin{aligned}\frac{\partial V}{\partial A}(S_i, A_{h+\frac{1}{2}}, t_{k+\frac{1}{2}}) &= \frac{1}{2}\Big(\frac{\partial V}{\partial A}(S_i, A_{h+\frac{1}{2}}, t_{k+1}) + \frac{\partial V}{\partial A}(S_i, A_{h+\frac{1}{2}}, t_k)\Big) + O(\Delta t^2)\\ &= \frac{V_{i,h+1}^{k+1} - V_{i,h}^{k+1}}{2\Delta A} + \frac{V_{i,h+1}^k - V_{i,h}^k}{2\Delta t} + O(\Delta t^2 + \Delta A^2)\,.\end{aligned}$$

After substituting the above approximations into the PDE and discarding the error terms, we get the following equations for the approximate values of the option prices:
(4.7)

$$\overline{a}_i V^k_{i-1,h} + \overline{b}_i V^k_{i,h} + \overline{c}_i V^k_{i+1,h} = \overline{d}_i V^{k+1}_{i-1,h+1} + \overline{e}_i V^{k+1}_{i,h+1} + \overline{f}_i V^{k+1}_{i+1,h+1} + \overline{g}_i (V^k_{i,h+1} - V^{k+1}_{i,h})$$

where, using the notation $\lambda = \frac{\Delta t}{\Delta S^2}$:

$$\begin{aligned}
&\overline{a}_i = \tfrac{\lambda}{2}\Big(- S_i^2\sigma^2 + rS_i\Delta S\Big)\,,\ \overline{b}_i = \Big(1 + \lambda S_i^2\sigma^2 + \tfrac{\log(S_i)\Delta t}{\Delta A} + r\Delta t\Big)\,,\ \overline{c}_i = -\tfrac{\lambda}{2}\Big(S_i^2\sigma^2 + rS_i\Delta S\Big)\,,\\
&\overline{d}_i = -a_i\,, \overline{e}_i = \Big(1 - \lambda S_i^2\sigma^2 + \tfrac{\log(S_i)\Delta t}{\Delta A} - r\Delta t\Big)\,, \overline{f}_i = -c_i\,, \overline{g}_i = \Big(- 1 + \tfrac{\log(S_i)\Delta t}{\Delta A}\Big)\,.
\end{aligned}$$

The procedure for solving the option pricing equation is as follows:

1. Fill the values $V^{N_t}_{i,h}$, $i = 0, \cdots, N_S, h = 0, \cdots, N_A$ using the payoff function.
2. For each $k = N_t - 1 : -1 : 0$
(a) Apply the boundary condition at $A = A_{\max}$ to define V^k_{i,N_A}, for $i = 0, \cdots, N_S$.
(b) For each $h = N_A - 1 : -1 : 0$, solve the three-diagonal system in the unknowns the values $V^k_{i,h}$, for $i = 1, \cdots, N_S - 1$, using the boundary condition at $S = S_{\min}$ and $S = B$.

This is a time-explicit difference scheme too. If we need the option price only at $t = 0$, then it is not necessary to store the full matrix V of approximate option prices; in fact we need only two levels, say V_{old} corresponding to $t = t_{k+1}$ and V_{new} corresponding to the current time level $t = t_k$. At the beginning V_{old} is computed using the final condition and at the end of each time step the values of V_{new} are copied to V_{old}. Anyway it requires the resolution of a linear system: the three-diagonal matrix $M = diag(\overline{a}, \overline{b}, \overline{c})$ can be assembled and factorized at the beginning, outside the cycles, depending only on the S-grid but its determinant is very high and it rises with the decreasing of ΔS causing numerical troubles.

5. Comparisons. Numerical examples concern the pricing problem of a call Asian option with an up-and-out barrier (2.9)-(2.12) and fixed strike payoff (2.4).

• *1st example*
In this example we use the same finance parameters found in the Release Notes of Matlab® R2017a in the section “Pricing Asian Options” (and shown in Table 1)[8], referring to the use of function `asianbykv` that provides the analytical solution to geometric Asian option that is available in Ref. [14] in the contract without barriers. We computed, at $t = 0$ and $A = 0$, a fixed strike call option approximation with up-and-out barrier at $S = 150 =: B$, by setting $A_{\min} = 0$, $A_{\max} \approx 5$, in accordance with (3.5) and (3.4), and $N_t = N_A = 20$. The results obtained by SABO are displayed in Fig. 2 on the left as function of S.

B	T	E	r	σ
150	1	90	0.035	0.2

Table 1: Fixed strike up-and-out call option data.

[8]These values of financial parameters are however usual in scientific articles: look at Refs. [6, 18, 19].

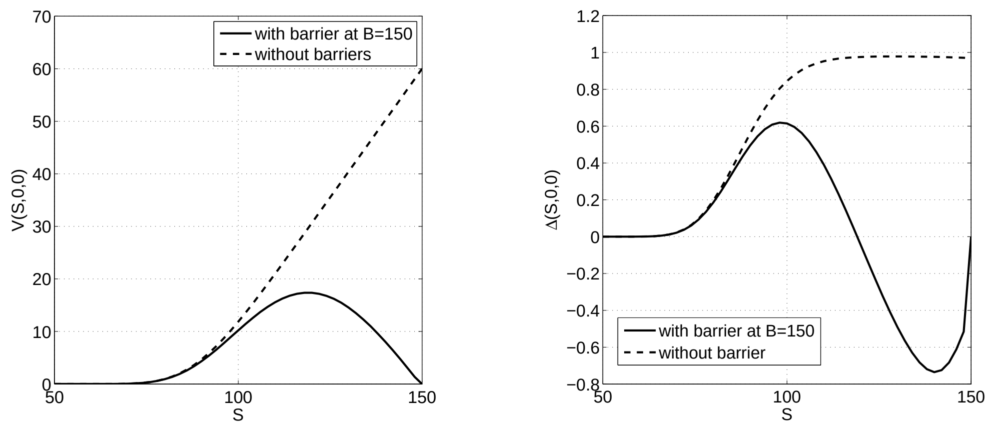

Fig. 2: Call up-and-out Geometric Asian option values and the associated $\mathbf{\Delta}$-values obtained by SABO.

We can observe that the solution appears to be smooth and, in compliance with previsions, it stays under the option value without barriers with a behavior analogous to that of European barrier options (look at Ref. [8]).
The same behavior is catched by the two proposed FD methods so, in order to discuss about efficiency and convergence, look at stabilization of digits in tables below.
For what concerns SABO, the stabilization of digits at $S = 100, 120, 140$ is fast (Tab. 2 on the left): at each level of mesh refinement (doubling of parameters N_t and N_A), with related computational time increasing (quadrupling), there is the achievement of about 1 digit of accuracy. As expected (Ref. [8]), the nearer the barrier the slower the convergence.

$V(S,0,0)$

$N_t = N_A$	$S = 100$	$S = 120$	$S = 140$	elapsed time (sec)
10	10.2170	17.3650	8.0877	$3.0 \cdot 10^0$
20	10.1480	17.2561	7.9929	$1.1 \cdot 10^1$
40	10.1419	17.2960	8.1400	$4.3 \cdot 10^1$
80	10.1432	17.3061	8.1507	$1.7 \cdot 10^2$
160	10.1438	17.3086	8.1551	$6.9 \cdot 10^2$
320	10.1439	17.3094	8.1566	$3.0 \cdot 10^3$
640	10.1440	17.3096	8.1570	$1.3 \cdot 10^4$

$\mathbf{\Delta}(S,0,0)$

$N_t = N_A$	$S = 100$	$S = 120$	$S = 140$
10	0.6214	−0.0491	−0.6895
20	0.6149	−0.0441	−0.7356
40	0.6142	−0.0383	−0.7643
80	0.6144	−0.0380	−0.7733
160	0.6144	−0.0379	−0.7742
320	0.6144	−0.0378	−0.7742
640	0.6144	−0.0378	−0.7742

Table 2: $V(S,0,0)$ and $\mathbf{\Delta}(S,0,0)$ evaluated by SABO at $S = 100, 120, 140$.

Moreover we can straightforwardly compute $\mathbf{\Delta}$ (Tab. 2 on the right) by formula (3.16) (global behavior displayed in Fig. 2 on the right). Some reference values to check the reliability of formula (3.16) can be given by the approximation of $\mathbf{\Delta}$ with the fourth order centered formula

(5.1)

$$\mathbf{\Delta} = \frac{-V(S+2\Delta S,0,0)+8V(S+\Delta S,0,0)-8V(S-\Delta S,0,0)+V(S-2\Delta S,0,0)}{12\Delta S}$$

applied to SABO values V computed by $N_t = N_A = 320$ and reducing ΔS. These values gives us also an idea of the refinement level of S-grid to be used in finite difference methods (Tab. 3).

$\mathbf{\Delta}(S,0,0)$ by formula (5.1)

ΔS	$S=100$	$S=120$	$S=140$
8	0.6103	-0.0375	-0.8209
4	0.6142	-0.0378	-0.7741
2	0.6144	-0.0378	-0.7742
1	0.6144	-0.0378	-0.7742

Table 3: $\mathbf{\Delta}(S,0,0)$ evaluated by formula (5.1) at $S=100,120,140$.

For what concerns FD1, the chosen discretization parameters are $\Delta t = 10^{-3}/2^k$, $\Delta A = 10^{-2}/2^k$ and, for the discretization of $S-$interval $(0,B)$, either $\Delta S = 2$ or $\Delta S = 1$. The results are reported respectively in Tabs. 4 and 5.

$V(S,0,0)$

k	$S=100$	$S=120$	$S=140$	elapsed time (sec)
1	11.7620	18.0788	8.4291	$3.7\cdot 10^0$
2	10.9704	17.6711	8.2893	$3.2\cdot 10^1$
3	10.5597	17.4862	8.2243	$1.3\cdot 10^2$
4	10.3508	17.3993	8.1934	$5.2\cdot 10^2$
5	10.2454	17.3572	8.1784	$2.1\cdot 10^3$
6	10.1926	17.3364	8.1710	$8.2\cdot 10^3$
7	10.1661	17.3261	8.1673	$3.5\cdot 10^4$

$\mathbf{\Delta}(S,0,0)$

k	$S=100$	$S=120$	$S=140$
1	0.5621	-0.0649	-0.7993
2	0.5812	-0.0503	-0.7853
3	0.5948	-0.0436	-0.7789
4	0.6031	-0.0404	-0.7759
5	0.6076	-0.0389	-0.7744
6	0.6100	-0.0381	-0.7737
7	0.6112	-0.0378	-0.7733

Table 4: $V(S,0,0)$ evaluated by FD1 at $S=100,120,140$, with $\Delta S=2$.

$V(S,0,0)$

k	$S=100$	$S=120$	$S=140$	elapsed time (sec)
1	11.7656	18.0744	8.4242	$1.4\cdot 10^1$
2	10.9740	17.6666	8.2845	$5.4\cdot 10^1$
3	10.5632	17.4817	8.2195	$2.2\cdot 10^2$
4	10.3541	17.3947	8.1887	$9.4\cdot 10^2$
5	10.2488	17.3526	8.1737	$4.1\cdot 10^3$
6	10.1959	17.3319	8.1663	$1.7\cdot 10^4$
7	10.1694	17.3216	8.1626	$6.9\cdot 10^4$

$\mathbf{\Delta}(S,0,0)$

k	$S=100$	$S=120$	$S=140$
1	0.5634	-0.0651	-0.8002
2	0.5826	-0.0506	-0.7863
3	0.5962	-0.0438	-0.7799
4	0.6045	-0.0407	-0.7768
5	0.6091	-0.0392	-0.7754
6	0.6115	-0.0384	-0.7746
7	0.6127	-0.0381	-0.7743

Table 5: $V(S,0,0)$ evaluated by FD1 at $S = 100, 120, 140$, with $\Delta S = 1$.

The poor efficiency of FD1 w.r.t. SABO is evident observing that, with the same CPU time order of computation[9] SABO is much more accurate than FD1. Further, referring to FD1, note that the mesh refinement in S-domain is raw in fact, comparing Tabs. 4 and 5, we can observe that, doubling the S-mesh, the third decimal place for V is still not fixed although the time of computation has grown by one order of magnitude. The values of $\mathbf{\Delta}$ are doubtful because, when $\Delta S = 2$, results monotonically decrease refining Δt and ΔA but not towards the right convergence values as it is suggested by results in Tab. 4 lower than results in Tab. 5 with the refinement of ΔS.

For what concerns FD2, results are reported in Table 6 and 7, with $\Delta t = \Delta A = 0.1/2^{k+2}$, $\Delta S = 2$ and $\Delta S = 1$ respectively. Approximations of derivatives in t and A variables are both of second order accuracy, this is reflected in finer results achieved by FD2 w.r.t. FD1 where approximations are of first order only.

$V(S,0,0)$

k	$S = 100$	$S = 120$	$S = 140$	elapsed time (sec)
1	9.7634	15.9334	7.8061	$1.6 \cdot 10^0$
2	9.7112	17.2597	8.3164	$6.0 \cdot 10^0$
3	10.0357	17.3093	8.1742	$2.3 \cdot 10^1$
4	10.1157	17.3205	8.1677	$1.1 \cdot 10^2$
5	10.1344	17.3186	8.1658	$6.1 \cdot 10^2$
6	10.1386	17.3174	8.1647	$3.0 \cdot 10^3$
7	10.1395	17.3166	8.1641	$1.3 \cdot 10^4$

$\mathbf{\Delta}(S,0,0)$

k	$S = 100$	$S = 120$	$S = 140$
1	0.4751	-0.0028	-0.7297
2	0.5754	$+0.0001$	-0.7849
3	0.6115	-0.0352	-0.7734
4	0.6126	-0.0367	-0.7733
5	0.6126	-0.0372	-0.7732
6	0.6126	-0.0374	-0.7731
7	0.6125	-0.0374	-0.7730

Table 6: $V(S,0,0)$ evaluated by FD2 at $S = 100, 120, 140$, with $\Delta S = 2$.

Anyway also with FD2, accuracy is still influenced by the coarseness of S-grid and SABO is still more performing.

$V(S,0,0)$

k	$S = 100$	$S = 120$	$S = 140$	elapsed time (sec)
1	9.7672	15.9289	7.8006	$4.8 \cdot 10^0$
2	9.7159	17.2545	8.3118	$2.1 \cdot 10^1$
3	10.0396	17.3046	8.1694	$7.6 \cdot 10^1$
4	10.1191	17.3160	8.1630	$3.0 \cdot 10^2$
5	10.1377	17.3141	8.1611	$1.2 \cdot 10^3$
6	10.1419	17.3129	8.1599	$4.7 \cdot 10^3$
7	10.1428	17.3121	8.1594	$1.9 \cdot 10^4$

$\mathbf{\Delta}(S,0,0)$

k	$S = 100$	$S = 120$	$S = 140$
1	0.4762	-0.0029	-0.7306
2	0.5767	-0.0001	-0.7859
3	0.6129	-0.0354	-0.7743
4	0.6141	-0.0370	-0.7743
5	0.6141	-0.0375	-0.7741
6	0.6140	-0.0377	-0.7740
7	0.6140	-0.0377	-0.7739

[9] All the numerical simulations have been performed with a laptop computer: CPU Intel i5, 4Gb RAM. Moreover, the structure of SABO linear system is very suitable for parallelization that should lead to further fastening.

Table 7: $V(S,0,0)$ evaluated by FD2 at $S = 100, 120, 140$, with $\Delta S = 1$.

In order to optimize computational costs, in FD2, we tried to reduce the asset domain to an interval $(S_{\min}, B)$ with $S_{\min} > 0$ and to apply the boundary condition (4.4). Comparing the computational costs of Tabs. 6 and 8 (left), one can observe that the savings achieved by reduction of computational domain are completely canceled by the major calculation due to the evaluation of (4.4). Moreover comparing results in Tab. 8 (left and right), one can note that numerical results are quite sensible to the choice of an artificial left bound.

$V(S,0,0) \quad S \in (50, B)$

k	$S = 100$	$S = 120$	$S = 140$	elapsed time (*sec*)
1	9.7634	15.9334	7.8061	$6.4 \cdot 10^0$
2	9.7112	17.2597	8.3164	$2.3 \cdot 10^1$
3	10.0357	17.3093	8.1742	$9.2 \cdot 10^1$
4	10.1157	17.3205	8.1677	$3.8 \cdot 10^2$
5	10.1344	17.3186	8.1658	$1.6 \cdot 10^3$
6	10.1386	17.3174	8.1647	$6.8 \cdot 10^3$
7	10.1395	17.3166	8.1641	$2.8 \cdot 10^4$

$V(S,0,0) \quad S \in (70, B)$

k	$S = 100$	$S = 120$	$S = 140$	elapsed time (*sec*)
1	9.6510	15.9098	7.8023	$1.1 \cdot 10^0$
2	9.6629	17.2441	8.3141	$3.9 \cdot 10^0$
3	10.0147	17.2972	8.1719	$1.5 \cdot 10^1$
4	10.1048	17.3096	8.1653	$6.4 \cdot 10^1$
5	10.1266	17.3080	8.1634	$3.2 \cdot 10^2$
6	10.1317	17.3068	8.1622	$1.5 \cdot 10^3$
7	10.1328	17.3060	8.1617	$6.7 \cdot 10^3$

Table 8: $V(S,0,0)$ evaluated by FD2 at $S = 100, 120, 140$, with $\Delta S = 2$ over $(S_{\min}, B)$. On the left: $S_{\min} = 50$. On the right: $S_{\min} = 70$.

• *2nd example*
In this example, the barrier is set at $S = 110 = B$ and the finance parameters applied are taken from [19]:

B	T	E	r	σ
110	1	100	0.15	0.05

Table 9: Fixed strike up-and-out call option data.

This set of data is troubling because $r >> \sigma^2$ and, by consequence, the diffusive term in Eq. (2.9) is dominated by the transport term; this kind of situation is well know also in the simpler Black and Scholes context (Ref. [16]) as it emphasizes instabilities (look at Fig. 3) that can be handled by mesh refinements or weakening accuracy by artificial diffusive terms.

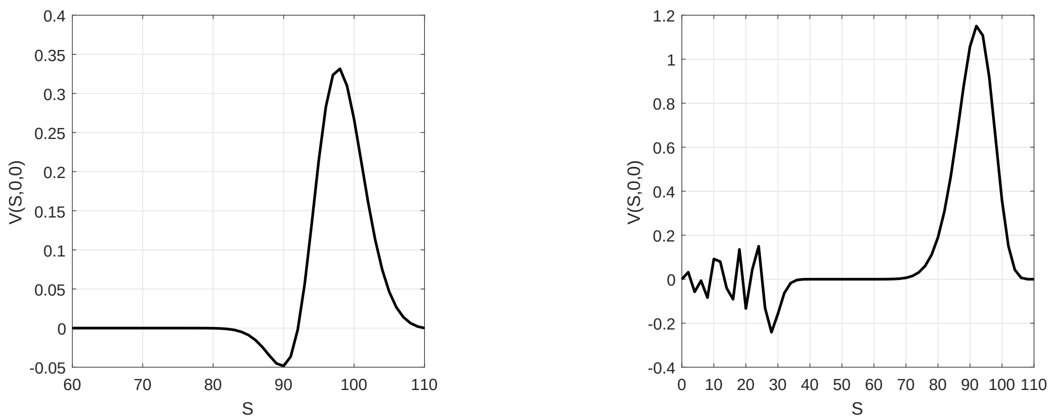

Fig. 3: On the left: numerical solution simulated by SABO with $N_t = N_A = 30$ with the appearance of a wrong oscillation. On the right: erroneous numerical solution simulated by FD2 with $N_t = 80$ and $N_A = 100$.

Also here $A_{\min} = 0$ and $A_{\max} \approx 5$, in accordance with (3.5) and (3.4). SABO needs a mesh of at least $N_t = N_A = 50$ basis functions to obtain a shape without oscillations (look at Fig. 4) but no other particular tricks.

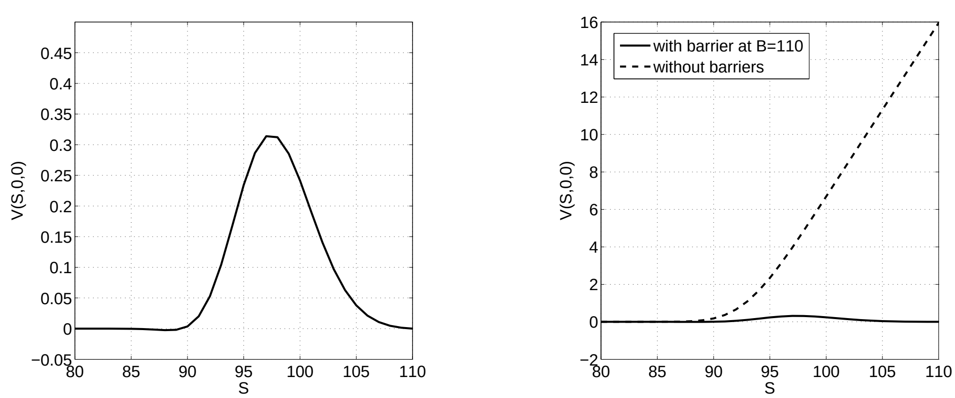

Fig. 4: On the left: correct geometric Asian up-and-out call option value obtained by SABO with data in Tab. 9. On the right: comparison between geometric Asian option values with up-and-out barrier and without barriers.

Concerning convergence, we can observe in Tab. 10 the stabilization of digits at $S = 90, 97, 104$ together with the CPU computational time in relation to the mesh refinement.

$N_t = N_A$	$S = 90$	$S = 97$	$S = 104$	elapsed time (sec)
50	0.0036	0.3138	0.0628	$3.4 \cdot 10^1$
100	0.0110	0.3291	0.0622	$1.4 \cdot 10^2$
200	0.0150	0.3299	0.0623	$5.7 \cdot 10^2$
400	0.0160	0.3300	0.0623	$2.4 \cdot 10^3$
800	0.0162	0.3301	0.0623	$2.5 \cdot 10^4$

Table 10: $V(S, 0, 0)$ evaluated by SABO at $S = 90, 97, 104$.

For what concerns FD1, results are reported in Tabs 11 and 12, where $\Delta t = 10^{-3}/2^k$ and $\Delta A = 10^{-2}/2^k$ for $\Delta S = 2$ and $\Delta S = 1$ respectively. Even if oscillations are not evident plotting the global solution as function of S, numerical results in Tab. 13 seem to be quite unstable and far from convergence: doubling the S-grid there is a substantial change in approximate solution.

k	$S = 90$	$S = 97$	$S = 104$	elapsed time (sec)
1	2.1459	1.5092	0.0795	$3.1 \cdot 10^0$
2	1.2189	1.0876	0.0645	$1.1 \cdot 10^1$
3	0.6548	0.8171	0.0563	$8.9 \cdot 10^1$
4	0.3332	0.6429	0.0522	$3.7 \cdot 10^2$
5	0.1668	0.5325	0.0505	$1.6 \cdot 10^3$
6	0.0884	0.4652	0.0499	$6.5 \cdot 10^3$
7	0.0532	0.4262	0.0496	$2.7 \cdot 10^4$

Table 11: $V(S, 0, 0)$ evaluated by FD1 at $S = 90, 97, 104$, with $\Delta S = 2$ (instead of $V(97, 0, 0)$, we have evaluated $(V(96, 0, 0) + V(98, 0, 0))/2$).

k	$S = 90$	$S = 97$	$S = 104$	elapsed time (sec)
1	2.1907	1.3826	0.1078	$5.8 \cdot 10^0$
2	1.2532	0.9922	0.0857	$4.9 \cdot 10^1$
3	0.6813	0.7425	0.0728	$2.0 \cdot 10^2$
4	0.3515	0.5823	0.0658	$8.2 \cdot 10^2$
5	0.1764	0.4814	0.0625	$3.3 \cdot 10^3$
6	0.0911	0.4206	0.0611	$1.4 \cdot 10^4$
7	0.0518	0.3861	0.0606	$5.3 \cdot 10^4$

Table 12: $V(S, 0, 0)$ evaluated by FD1 at $S = 90, 97, 104$, with $\Delta S = 1$.

Results reported in Tab. 13 refer to FD2, applied with $\Delta t = \Delta A = 0.1/2^{k+2}$ and

$\Delta S = 2$ and show the same bad behavior of FD1 scheme.

k	$S = 90$	$S = 97$	$S = 104$	elapsed time (sec)
1	2.1116	1.1155	0.0473	$1.3 \cdot 10^0$
2	1.0794	0.7553	0.0402	$4.4 \cdot 10^0$
3	0.5115	0.5410	0.0387	$1.7 \cdot 10^1$
4	0.2215	0.4199	0.0423	$8.1 \cdot 10^1$
5	0.0929	0.3672	0.0493	$4.1 \cdot 10^2$
6	0.0446	0.3641	0.0502	$2.0 \cdot 10^3$
7	0.0290	0.3761	0.0491	$8.8 \cdot 10^3$

Table 13: $V(S, 0, 0)$ evaluated by FD2 at $S = 90, 97, 104$ (instead of $V(97, 0, 0)$, we have evaluated $(V(96, 0, 0) + V(98, 0, 0))/2$), with $\Delta S = 2$.

However refining in S (for numerical results with $\Delta S = 1$ and $\Delta S = 0.5$, look at Tabs. 14 and 15 respectively), FD2 improves more rapidly towards results obtained by SABO that remains noticeably more efficient.

k	$S = 90$	$S = 97$	$S = 104$	elapsed time (sec)
1	2.1287	1.0161	0.0701	$2.6 \cdot 10^0$
2	1.0961	0.6858	0.0557	$1.0 \cdot 10^1$
3	0.5255	0.4894	0.0496	$4.0 \cdot 10^1$
4	0.2308	0.3785	0.0503	$1.6 \cdot 10^2$
5	0.0965	0.3303	0.0567	$7.0 \cdot 10^2$
6	0.0436	0.3290	0.0608	$2.8 \cdot 10^3$
7	0.0255	0.3427	0.0602	$1.1 \cdot 10^4$

Table 14: $V(S, 0, 0)$ evaluated by FD2 at $S = 90, 97, 104$, with $\Delta S = 1$.

k	$S = 90$	$S = 97$	$S = 104$	elapsed time (sec)
1	2.1305	0.9954	0.0744	$7.7 \cdot 10^0$
2	1.0990	0.6696	0.0586	$3.0 \cdot 10^1$
3	0.5285	0.4760	0.0515	$1.2 \cdot 10^2$
4	0.2331	0.3665	0.0517	$6.4 \cdot 10^2$
5	0.0975	0.3186	0.0578	$3.6 \cdot 10^3$
6	0.0435	0.3165	0.0624	$1.8 \cdot 10^4$
7	0.0246	0.3294	0.0619	$1.8 \cdot 10^5$
8	0.0188	0.3334	0.0619	$6.6 \cdot 10^5$
9	0.0172	0.3343	0.0620	$3.3 \cdot 10^6$

Table 15: $V(S, 0, 0)$ evaluated by FD2 at $S = 90, 97, 104$, with $\Delta S = 0.5$.

• *3rd example*

In this example, the barrier is set at $S = 1 = B$ and the finance parameters applied are:

B	T	E	r	σ
1	1	0.5	0.035	0.4

Table 16: Fixed strike up-and-out call option data.

Note that considering $B = 1$ implies $\mathrm{A}_{\min} \approx -1$ and $\mathrm{A}_{\max} \approx 1$ (according to (3.5) and (3.4)) and asset values lower than 1. The solution obtained by SABO with $\Delta t = 0.1$ and $\Delta A = 0.2$ is plotted in Fig. 5 together with the analytical solution of the fixed strike call option without barriers.

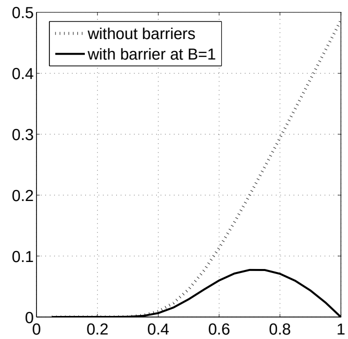

Fig. 5: Comparison between geometric Asian option values with up-and-out barrier and without barriers.

In Tab. 17 we can observe the stabilization of digits at $S = 0.5, 0.7, 0.9$ together with the CPU computational time in relation to the mesh refinement.

$N_t = N_A$	$S = 0.5$	$S = 0.7$	$S = 0.9$	elapsed time (sec)
10	$0.2911 \cdot 10^{-1}$	$0.7788 \cdot 10^{-1}$	$0.4693 \cdot 10^{-1}$	$1.5 \cdot 10^{0}$
20	$0.2918 \cdot 10^{-1}$	$0.7730 \cdot 10^{-1}$	$0.4275 \cdot 10^{-1}$	$5.8 \cdot 10^{0}$
40	$0.2919 \cdot 10^{-1}$	$0.7736 \cdot 10^{-1}$	$0.4114 \cdot 10^{-1}$	$2.7 \cdot 10^{1}$
80	$0.2919 \cdot 10^{-1}$	$0.7739 \cdot 10^{-1}$	$0.4058 \cdot 10^{-1}$	$1.5 \cdot 10^{2}$
160	$0.2919 \cdot 10^{-1}$	$0.7740 \cdot 10^{-1}$	$0.4040 \cdot 10^{-1}$	$9.4 \cdot 10^{2}$
320	$0.2919 \cdot 10^{-1}$	$0.7740 \cdot 10^{-1}$	$0.4035 \cdot 10^{-1}$	$2.2 \cdot 10^{3}$

Table 17: $V(S, 0, 0)$ evaluated by SABO at $S = 0.5, 0.7, 0.9$.

In order to check numerical results, we consider only FD1 method with approximation of A derivative by backward difference because, as suggested in Sec. 4.1, if $\log(S) < 0$ the information moves backward in time from the left bound of A-domain. Results in Tab. 18 are obtained for $A_{\min} = -1$, $A_{\max} = 0$, $\Delta t = 10^{-3} \cdot 2^{-k}$, $\Delta A = 10^{-2} \cdot 2^{1-k}$ and $\Delta S = 0.01$.

k	$S = 0.5$	$S = 0.7$	$S = 0.9$	elapsed time (sec)
1	$0.3247 \cdot 10^{-1}$	$0.7923 \cdot 10^{-1}$	$0.4075 \cdot 10^{-1}$	$7.0 \cdot 10^{-1}$
2	$0.3084 \cdot 10^{-1}$	$0.7833 \cdot 10^{-1}$	$0.4055 \cdot 10^{-1}$	$1.9 \cdot 10^{0}$
3	$0.3002 \cdot 10^{-1}$	$0.7787 \cdot 10^{-1}$	$0.4045 \cdot 10^{-1}$	$7.2 \cdot 10^{0}$
4	$0.2960 \cdot 10^{-1}$	$0.7765 \cdot 10^{-1}$	$0.4040 \cdot 10^{-1}$	$2.9 \cdot 10^{1}$
5	$0.2940 \cdot 10^{-1}$	$0.7753 \cdot 10^{-1}$	$0.4038 \cdot 10^{-1}$	$2.6 \cdot 10^{2}$
6	$0.2929 \cdot 10^{-1}$	$0.7747 \cdot 10^{-1}$	$0.4037 \cdot 10^{-1}$	$1.1 \cdot 10^{3}$

Table 18: $V(S,0,0)$ evaluated by FD1 at $S = 0.5, 0.7, 0.9$, with $\Delta S = 0.01$.

6. Conclusions. The comparisons between SABO method and some basic finite difference methods illustrate the efficiency of SABO.
SABO allows to avoid inaccurate choice of boundary conditions and tricky evaluations of fictitious bounds for the computational domain. SABO proves to be stable with respect to the variation of financial parameters and discretization parameters; on the contrary, we noted a particular difficulty in choosing a starting triad $(\Delta S, \Delta A, \Delta t)$ rightly calibrated for running simulations with FD methods whose stability is very sensitive to the variation of the discretization parameters. Furthermore, SABO appears to have higher accuracy in computing both option values and Greeks.

7. Acknowledgments. The authors gratefully acknowledge use of a GNCS-INdAM grant for the dissemination of these results.

REFERENCES

[1] A. Aimi and C. Guardasoni, *Collocation Boundary Element Method for the pricing of Geometric Asian Options*, Eng. Anal. Bound. Elem, in press.
[2] C. Atkinson and S. Kazantzaki, *Double knock-out Asian barrier options which widen or contract as they approach maturity*, Quant. Finance, 9 (2009), pp. 329–340.
[3] L.V. Ballestra and G. Pacelli, *A boundary element method to price time-dependent double barrier options*, Appl. Math. Comput., 218 (2011), pp. 4192 – 4210.
[4] ———, *A very fast and accurate boundary element method for options with moving barrier and time-dependent rebate*, Appl. Numer. Math., 77 (2014), pp. 1 – 15.
[5] E. Barucci, S. Polidoro, and V. Vespri, *Some results on partial differential equations and Asian options*, Math. Models Methods Appl. Sci., 11 (2001), pp. 475–497.
[6] P. Boyle and A. Potapchik, *Prices and sensitivities of Asian options: a survey*, Insurance Math. Econom., 42 (2008), pp. 189–211.
[7] C. Guardasoni, *Semi-Analytical method for the pricing of Barrier Options in case of time-dependent parameters (with Matlab codes)*, Commun. Appl. Ind. Math., 8 (2018), pp. 42–67.
[8] C. Guardasoni and S. Sanfelici, *A boundary element approach to barrier option pricing in Black-Scholes framework*, Int. J. Comput. Math., 93 (2016), pp. 696–722.
[9] C. Guardasoni and S. Sanfelici, *Fast Numerical Pricing of Barrier Options under Stochastic Volatility and Jumps*, SIAM J. Appl. Math., 76 (2016), pp. 27–57.
[10] Lars Hörmander, *Hypoelliptic second order differential equations*, Acta Math., 119 (1967), pp. 147–171.
[11] J. Hugger, *A fixed strike Asian option and comments on its numerical solution*, ANZIAM J, 45 (2003/04), pp. C215–C231.
[12] ———, *Wellposedness of the boundary value formulation of a fixed strike Asian option*, J. Comput. Appl. Math, 185 (2006), pp. 460–481.
[13] R. Kangro, *Lecture notes on computational finance*, tech. report, University of Tartu, Fall 2011.
[14] A.G.Z. Kemna and A.C.F. Vorst, *A pricing method for options based on average asset values*, J. Bank. Financ, 14 (1990), pp. 113–129.
[15] S. Polidoro, *Uniqueness and representation theorems for solutions of Kolmogorov-Fokker-Planck equations*, Rend. Mat. Appl. VII, 15 (1995), pp. 535–560 (1996).
[16] Rüdiger U. Seydel, *Tools for computational finance*, Universitext, Springer-Verlag, Berlin, fourth ed., 2009.
[17] P. Wilmott, *Derivatives: the theory and practice of financial engineering*, John Wiley and Sons, 2000.
[18] J. E. Zhang, *Theory of continuously-sampled asian option pricing*, working papers, East Asian Bureau of Economic Research, 2004.
[19] R. Zvan, P. A. Forsyth, and K. Vetzal, *Robust numerical methods for pde models of asian options*, tech. report, Cheriton School of Computer Science, University of Waterloo, 1996.