Adaptive Finite Element Methods for Local Volatility European Option Pricing

Alexandre Ern
CERMICS, Ecole nationale des ponts et chaussées
6 et 8 avenue Blaise Pascal
77455 Marne la Vallée cedex 2, France
ern@cermics.enpc.fr

Stéphane Villeneuve
Université d'Evry Val d'Essonne
Equipe d'Analyse et de Probabilité
Boulevard F.Mitterand
91 025 Evry cedex, France
svillene@maths.univ-evry.fr

Antonino Zanette
Dipartimento di Finanza dell'Impresa
e dei Mercati Finanziari
Università di Udine
via Tomadini 30/A, Udine, Italy
antonino.zanette@dfimf.uniud.it

October 31, 2002

Abstract

We investigate finite element discretizations using functions that are discontinuous in time and continuous in space for European options with local volatility Black-Scholes models. We present an a posteriori error estimate where a user-specified functional of the error is controlled by the inner product of the finite element residual with the solution of a dual problem that involves the density of the target functional as prescribed data. Examples of error functionals are discussed in the context of either option pricing or volatility calibration from market data. The a posteriori error estimator is then localized onto the space-time cells of the computational mesh and implemented in the framework of an adaptive mesh refinement/derefinement algorithm which provides some form of optimal compromise between accuracy requirements and computational costs. Numerical examples illustrate the efficiency of the proposed methodology.

1 Introduction

Over the last few years, significant progress has been made in the understanding of adaptive finite element approximation of partial differential equations (PDEs) based on a posteriori error estimation. For a recent review including residual error estimators, error estimators based on the solution of local problems and hierarchical basis error estimators, we refer to [13]. Among a posteriori error estimators, a promising approach appears to be the dual weighted residual (DWR) method. Its key advantage is that it allows to control the error by means of a user-specified functional output that may target quantities of interest. In the context of finance problems, this is particularly appealing for pricing problems where the practitioner needs to compute highly accurate option prices for specific values of the stock. Targeting quantities of interest may also be extremely useful in calibration problems where the volatility map is reconstructed from market prices. Another advantage of the DWR method is

that error propagation through the computational domain is accounted for via the solution of a dual problem whose data is the density of the prescribed functional. From a financial viewpoint, this is particularly important for low volatility problems where the hyperbolic part of the PDE may dominate the elliptic terms.

The main concept in the DWR method is to introduce an auxiliary PDE problem, usually called the dual problem and written in terms of the formal adjoint of the PDE under consideration. The a posteriori error estimator may then be expressed in terms of various inner products involving the finite element residual of the numerical solution and quantities depending on the dual solution. Theoretical results concerning the DWR method are presented in [5, 6, 2] for steady and unsteady problems. The DWR method may be conveniently implemented in the framework of an adaptive mesh refinement/derefinement procedure. To this purpose, the inner products in the error estimator are first localized into the space-time cells of the computational mesh. Local element bounds are then used to decide whether to refine (or derefine) the mesh locally. The efficiency of the DWR method for an extensive range of engineering problems is highlighted in [2].

The aim of this paper is to investigate the effectiveness of the DWR method in order to solve the partial differential equations associated with European option pricing in local volatility Black-Scholes models. In the Black-Scholes model, the price of a dividend-paying stock S_t follows the stochastic differential equation

$$\frac{dS_t}{S_t} = (r-\delta)dt + \sigma dB_t, \tag{1}$$

where the constant σ is the volatility of the stock. It is well-known that the price of a European call option has a closed-form solution. Nevertheless, since the volatility of the stock is not directly observable, practitioners often invert the closed-form solution in order to find the volatility σ (usually called *implied volatility*) that yields the best agreement with the market option price. It is well-known that the implied volatility varies with the strike and the time - *the smile effect*. Therefore, the model has to be extended in order to take this phenomenon into account. A type of model called *local volatility* considers that the volatility is a deterministic function of time and the underlying asset.

In this paper, we consider a market model of local volatility where the evolution of the stock-price is governed by the stochastic differential equation

$$\frac{dS_t}{S_t} = (r-\delta)dt + \sigma(t,S_t)dB_t, \tag{2}$$

where the interest rate r and the dividend rate δ are nonnegative constants and $(B_t)_{0\leq t\leq T}$ is a standard Brownian motion. The volatility σ is a $C^{1,2}([0,T]\times\mathbb{R}_+)$ function and is assumed to be uniformly bounded, i.e. there exist two positive constants $\underline{\sigma},\overline{\sigma}$ such that $\overline{\sigma}\geq\sigma(t,x)\geq\underline{\sigma}>0$ for every $(t,x)\in[0,T]\times\mathbb{R}_+$.

Consider a derivative security with terminal payoff $\psi(S_T)$, where ψ is some continuous real function. In the absence of arbitrage, the price of the derivative is given by $P(t,S_t)$ where P solves the partial differential equation

$$\begin{cases} P_t + \mathcal{A}P = 0 & \text{on } (0,T]\times\mathbb{R}_+, \\ P(T,\cdot) = \psi & \text{on } \mathbb{R}_+, \end{cases} \tag{3}$$

with

$$\mathcal{A}P(t,x) = \hat{\sigma}(t,x)P_{xx}(t,x) + (r-\delta)xP_x(t,x) - rP(t,x), \tag{4}$$

and $\hat{\sigma}(t,x) = x^2\frac{\sigma^2(t,x)}{2}$. In the context of PDEs, subscripts t and x refer to partial derivatives with respect to time and space respectively.

There are two problems in practice. One is the pricing problem which consists in computing option prices with reliable accuracy. Such problems are known to be difficult, especially near maturity. The second one is the calibration problem which focuses on the reconstruction of the volatility map from market prices. This paper will mainly focus on option pricing problems. For calibration problems, we refer to a recent paper [3] and references therein.

The paper is organized as follows. In Section 2, the finite element discretization with discontinuous in time and continuous in space functions is made concrete for European option problems with local volatility Black-Scholes models. The a posteriori error estimates are derived in Section 3. The adaptive mesh refinement algorithm including practical implementation is discussed in Section 4. Finally, numerical results are presented in Section 5.

2 Discretization by space-time finite elements

We are interested in the numerical computation of the price function P. The numerical procedure consists in the following three steps:

- the parabolic problem (3) is localized to a bounded domain in space;
- the localized problem is written in weak form;
- an approximate solution is sought by means of a non-conforming Galerkin method involving discrete functions that are discontinuous in time and continuous in space.

In the sequel, it will be convenient to reverse the time variable and consider $u(t,\cdot) = P(T-t,\cdot)$.

2.1 Localization to a bounded domain

Consider the following approximation problem

$$\begin{cases} (u_a)_t - \mathcal{A}u_a = 0 \text{ on } (0,T]\times\Omega_a, \\ u_a(t,1/a) = 0 \text{ and } u_a(t,a) = C_a(t), \\ u_a(0,\cdot) = \psi \text{ on } \Omega_a, \end{cases} \tag{5}$$

where $\Omega_a = (1/a, a)$ and $C_a(t)$ is an artificial boundary value imposed at $x = a$ and that may depend on time. The choice of a must answer two main purposes. First, the most probable values of S_T have to be contained in $(1/a, a)$. Second, a must be high enough to ensure the convergence of the approximated value u_a to the option value u. In order to impose relevant boundary conditions, it is necessary to understand the behavior of the solution at infinity and near zero. It is well-known that the convergence when we let the domain tend to $(0,+\infty)$ is governed by a phenomenon of large deviation type [1] and therefore the choice of Dirichlet boundary conditions leads to an exponential error. More precisely, we have the following lemma.

Lemma 1. *Suppose there exist constants c_1 and c_2 such that $\psi(x) \le c_1\, x$ for $x \in \mathbb{R}_+$, and $C_a(t) \le c_2\, a$ for $t \in (0,T)$. Then for every $a > 0$ and every $(t,x) \in (0,T)\times\Omega_a$, we have*

$$|u(t,x) - u_a(t,x)| \le c_3\, a \exp\left(-\frac{(\log(\frac{a}{x}) - (\frac{\overline{\sigma}^2}{2} + (r-\delta))(T-t))^2}{2\overline{\sigma}^2(T-t)}\right),$$

with $c_3 = \max(c_1, c_2)$.

Proof. We give a short proof for $t = T$. Feynman-Kac formula yields

$$|u(T,x) - u_a(T,x)| \leq c_3 \Big(\mathbb{E}(e^{-rT} S_T^x \mathbf{1}_{\{\exists t \in [0,T] \,|\, S_t^x \notin]1/a, a[\}}) + a\mathbb{P}(\exists t \in [0,T] \,|\, S_t^x \notin]1/a, a[) \Big).$$

Since the function σ is bounded, we will define the probability

$$\frac{dQ}{d\mathbb{P}}|\mathcal{F}_t = \exp\left(\int_0^T \sigma(s, X_s)\, dB_s - \frac{1}{2} \int_0^T \sigma^2(s, X_s)\, ds \right).$$

Setting $X_t = \log S_t$ and $\lambda = \log a$, we get

$$|u(T,x) - u_a(T,x)| \leq c_3 a \left(\mathbb{P}(\sup_{0 \leq t \leq T} |X_t^{\log x}| \geq \lambda) + Q(\sup_{0 \leq t \leq T} |X_t^{\log x}| \geq \lambda) \right).$$

We only prove the exponential estimate for $\mathbb{P}(\sup_{0 \leq t \leq T} |X_t^{\log x}| \geq \lambda)$ since the same techniques apply for $Q(\sup_{0 \leq t \leq T} |X_t^{\log x}| \geq \lambda)$ with the $Q-$ Brownian motion

$$W_t := B_t - \int_0^T \sigma^2(s, X_s)\, ds.$$

For $\rho > 0$, let us introduce the martingale

$$M_t^\rho = \exp\left(\rho \left(X_t^{\log x} - \log x - \int_0^T b(s, X_s)\, ds \right) - \frac{\rho^2}{2} \int_0^T \sigma^2(s, X_s)\, ds \right).$$

Thus, using standard martingale inequalities,

$$\begin{aligned}
\mathbb{P}(\sup_{0 \leq t \leq T} |X_t^{\log x}| \geq \lambda) &\leq \mathbb{P}\left(\sup_{0 \leq t \leq T} |M_t^\rho| \geq \exp\left(\rho \left(\lambda - \log x - (\frac{\overline{\sigma}^2}{2} + (r - \delta))T \right) - \frac{\rho^2}{2} \int_0^T \sigma^2(s, X_s)\, ds \right) \right) \\
&\leq \mathbb{P}\left(\sup_{0 \leq t \leq T} |M_t^\rho| \geq \exp\left(\rho \left(\lambda - \log x - (\frac{\overline{\sigma}^2}{2} + (r - \delta))T \right) - \frac{\rho^2 \overline{\sigma}^2}{2} T \right) \right) \\
&\leq \exp\left(-\rho \left(\lambda - \log x - (\frac{\overline{\sigma}^2}{2} + (r - \delta))T \right) + \frac{\rho^2 \overline{\sigma}^2}{2} T \right)
\end{aligned}$$

We close the proof by choosing $\rho = \dfrac{\lambda - \log x - (\frac{\overline{\sigma}^2}{2} + (r - \delta))T}{\overline{\sigma} T}$. □

It is worthwhile to point out that a high value for a induces substantial numerical costs. This drawback is in part circumvented by the adaptive method as we will see later. In our numerical experiments, we shall consider European call options with $\psi(x) = (x - K)_+$ and therefore take $C_a(t) = ae^{-\delta t} - Ke^{-rt}$.

2.2 Weak formulation

In order to write the localized problem (5) in weak form, we introduce the functional space

$$W(0,T; H_0^1(\Omega_a)) = \left\{ \, u \in L^2(0,T; H_0^1(\Omega_a)); \; u_t \in L^2(0,T; H^{-1}(\Omega_a)) \, \right\}.$$

Owing to the trace property $W(0,T;H_0^1(\Omega_a)) \subset C^0(0,T;L^2(\Omega_a))$ with continuous injection (see [11]), we have $u(t,\cdot) \in L^2(\Omega_a)$ for all $t \in [0,T]$. The weak formulation of (5) reads

$$\left\{ \begin{array}{l} \text{find } u \in W(0,T;H_0^1(\Omega_a)) \text{ such that} \\ \displaystyle\int_0^T \langle u_t, v\rangle_{-1,1}\, dt + \int_0^T a(u,v)\, dt = \int_0^T (f,v)_{\Omega_a}\, dt, \quad \forall v \in W(0,T;H_0^1(\Omega_a)), \\ u(0,\cdot) = \psi \quad \text{in } L^2(\Omega_a), \end{array} \right. \tag{6}$$

where $\langle\cdot,\cdot\rangle_{-1,1}$ denotes the duality pairing between $H^{-1}(\Omega_a)$ and $H_0^1(\Omega_a)$ and $(\cdot,\cdot)_{\Omega_a}$ the inner product of $L^2(\Omega_a)$. Furthermore, the bilinear form $a(\cdot,\cdot)$ is given by

$$a(u,v) = (u_x, (\hat{\sigma} v)_x)_{\Omega_a} - (r-\delta)(x u_x, v)_{\Omega_a} + r(u,v)_{\Omega_a}.$$

Non-homogeneous Dirichlet boundary conditions at $x = a$ have been treated by introducing an appropriate right-hand side f in (6). f depends on t and x and without any loss of generality, one may choose $f \in C^\infty((0,T)\times\Omega_a)$.

The assumptions on σ ensure that the bilinear form a is strongly elliptic. Therefore, since the initial data ψ is in $H^1(\Omega_a)$ and the right-hand side f is smooth, problem (6) has a unique solution u which belongs to $C^1(0,T;H_0^1(\Omega_a))$ [11]. In the sequel, the duality pairing $\langle\cdot,\cdot\rangle_{-1,1}$ will thus be replaced by the $L^2(\Omega_a)$ inner product.

2.3 The non-conforming Galerkin method

A first approach that might be considered to discretize (6) is to combine finite difference schemes in time with a conforming finite element method in space. For a numerical analysis of associated schemes, we refer for instance to [7]. In this paper, we consider a different approach involving space-time finite elements [9, 5, 6]. More specifically, an approximate solution of (6) is sought by means of a non-conforming Galerkin method involving discrete functions that are discontinuous in time and continuous in space. The time interval $[0,T]$ is split into subintervals $I_n = (t_{n-1}, t_n]$ of length $k_n = t_n - t_{n-1}$, where $0 = t_0 < \ldots < t_N = T$. We denote by $S_n = I_n \times \Omega_a$ the time slab associated with I_n. In each time slab S_n, we consider a mesh $\mathcal{M}_n$ of the domain Ω_a consisting of $M_n + 1$ subintervals $K_i^n = (x_i^n, x_{i+1}^n)$ of length $h_i^n = x_i^n - x_{i-1}^n$ where $1/a = x_0^n < \ldots < x_{M_n+1}^n = a$. The space-time mesh is illustrated in figure 1.

For an integer $p \geq 1$, we denote by $P_c^p(\mathcal{M}_n)$ the space of continuous functions in space that are polynomials of degree $\leq p$ on each subinterval K_i^n. In particular, we shall consider the space $P_c^1(\mathcal{M}_n)$ spanned by the functions $(\phi_i^n)_{1\leq i\leq M_n}$ given by

$$\phi_i^n(x) = \left\{ \begin{array}{ll} \frac{x - x_{i-1}^n}{h_i^n} & \text{if } x_{i-1}^n \leq x \leq x_i^n, \\ \\ \frac{x_{i+1}^n - x}{h_{i+1}^n} & \text{if } x_i^n \leq x \leq x_{i+1}^n, \\ \\ 0 & \text{otherwise.} \end{array} \right.$$

For an integer $q \geq 0$, we define the space-time finite element space $P_{d/c}^{q,p}$ by

$$P_{d/c}^{q,p} = \left\{ v(t,x) = \sum_{n=1}^{N} \left(\sum_{l=0}^{q} v_l^n(x) \left(\frac{t - t_{n-1}}{k_n} \right)^l \right) \mathtt{1}_{I_n}(t) \text{ with } v_l^n \in P_c^p(\mathcal{M}_n) \right\}.$$

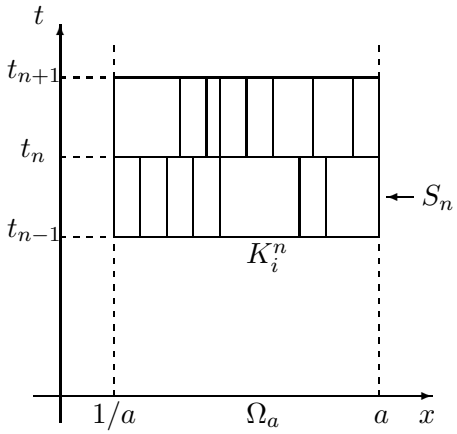

Figure 1: Space-time finite element mesh

Thus, on each cell $I_n \times K_i^n$, $v \in P_{d/c}^{q,p}$ is a polynomial of degree $\leq q$ in time whose coefficients are polynomials of degree $\leq p$ in space. Note that functions in $P_{d/c}^{q,p}$ are continuous in space but discontinuous in time. Therefore, $P_{d/c}^{q,p}$ is a non-conforming approximation space for $W(0,T;H_0^1(\Omega_a))$. For functions in $P_{d/c}^{q,p}$, the jumps occurring at each discrete time t_n ($n \geq 1$) are functions depending on the spatial variable x denoted by

$$\forall x \in \Omega_a, \qquad [v]_n(x) = v_n^+(x) - v_n^-(x) = \lim_{s \to 0^+} v(t_n + s, x) - v(t_n, x),$$

and for $n = 0$, we adopt the convention that $[v]_0 = v_0^+$.

We may now write the discretization of (6) as follows

$$\begin{cases} \text{find } U \in P_{d/c}^{q,p} \text{ such that} \\ B(U,V) = F(V), \quad \forall V \in P_{d/c}^{q,p}, \end{cases} \tag{7}$$

with the bilinear form $B(\cdot,\cdot)$

$$B(v,w) = \sum_{n=1}^{N} \int_{I_n} \left\{ (\tfrac{\partial v}{\partial t}, w)_{\Omega_a} + a(v,w) \right\} dt + \sum_{n=1}^{N} ([v]_{n-1}, w_{n-1}^+)_{\Omega_a}, \tag{8}$$

and the linear form $F(\cdot)$

$$F(w) = \int_0^T (f,w)_{\Omega_a} \, dt + (i_0\psi, w_0^+)_{\Omega_a}. \tag{9}$$

Here, $i_0\psi$ is (for instance) the L^2-projection of the terminal payoff function ψ onto the discrete space $P_c^p(\mathcal{M}_1)$. Since in general ψ is piecewise linear on the initial mesh $\mathcal{M}_1$, we simply have $i_0\psi = \psi$, an assumption that will be kept in the rest of this work. Notice that the initial condition $U(0,\cdot) = i_0\psi$ is readily recovered from (7). We also point out that in the case $i_0\psi = \psi$, the discrete problem (7) is consistent, i.e. the exact solution u of (6) also satisfies (7).

From a computational viewpoint, we shall focus on linear interpolation in space ($p = 1$) and either constant or linear interpolation in time ($q = 0$ or $q = 1$). For $1 \leq n \leq N$, let A_n be the discrete operator acting on $P_c^1(\mathcal{M}_n)$ such that $(A_nU, V)_{\Omega_a} = a(U,V)$ for $(U,V) \in P_c^1(\mathcal{M}_n)$ and let π_n be the L^2-projection operator onto $P_c^1(\mathcal{M}_n)$.

- for $q = 0$, set $U_n = U|_{I_n} \in P_c^1(\mathcal{M}_n)$ for $n \geq 1$. System (7) may then be recast into the form

$$U_n - \pi_n U_{n-1} + k_n A_n U_n = \int_{I_n} \pi_n f \, dt, \qquad n \geq 1, \tag{10}$$

  with initial condition $U_0 = i_0\psi$. For each time step, (10) is thus equivalent to the resolution of a backward Euler scheme.

- for $q = 1$, set $U_n = U_n^0 + \left(\frac{t - t_{n-1}}{k_n}\right) U_n^1$ for $n \geq 1$ with U_n^0 and $U_n^1 \in P_c^1(\mathcal{M}_n)$. System (7) then reduces to the following system of equations

$$\left\{ \begin{array}{l} (I + k_n A_n)U_n^0 + (I + \frac{k_n}{2} A_n)U_n^1 = \pi_n U_{n-1}^- + \displaystyle\int_{I_n} \pi_n f \, dt, \\ \frac{k_n}{2} A_n U_n^0 + (\frac{I}{2} + \frac{k_n}{3} A_n)U_n^1 = \displaystyle\int_{I_n} (t - t_{n-1})\pi_n f \, dt, \end{array} \right. \qquad n \geq 1, \tag{11}$$

  with initial condition $U_0^- = i_0\psi$.

Let u be the solution of the continuous problem (6) and U be the solution of the discrete problem (7) for $p = 1$ and $q = 0$ or 1. Then, under reasonable assumptions on the time steps and the regularity of the exact solution u, a priori error estimates show that the error $\max_{t\in[0,T]} \|u - U\|_{L^2(\Omega_a)}$ is of order 2 in space and $q + 1$ in time (see [5] for details). In the sequel, method (10) will be termed the dG(0) method and method (11) the dG(1) method.

3 A posteriori error analysis

In this section we briefly present the mathematical analysis of the a posteriori error estimator that will serve as the basis for the adaptive mesh refinement algorithm described in Section 4.

3.1 Output functionals of financial interest

Let $e = u - U$ be the error. Given two functionals $\theta_1 := \theta_1(t, x)$ for $(t, x) \in (0, T) \times \Omega_a$ and $\theta_2 := \theta_2(x)$ for $x \in \Omega_a$, our goal is to control the θ-error measure given by

$$\Theta(e) := \int_0^T (\theta_1, e)_{\Omega_a} \, dt + (\theta_2, e(T, \cdot))_{\Omega_a}. \tag{12}$$

The functionals θ_1 and θ_2 are user-specified functionals designed to target quantities of financial interest.

For instance, in the rather simple situation of an option pricing problem at a given strike value K, high demands on accuracy do not concern $e(t, x)$ for $(t, x) \in (0, T) \times \Omega_a$ but only $e(T, x)$ for $x \in \omega$ where ω is a small neighborhood of K. Therefore, an appropriate θ-error measure may be obtained by taking $\theta_1 = 0$ and

$$\theta_2(x) = \tfrac{1}{\|\sqrt{\psi} e(T,\cdot)\|_{L^2(\Omega_a)}} \, \psi(x) \, e(T, x),$$

where $\psi := \psi(x)$ is a function chosen by the user with support in ω. The θ-error measure simply reads

$$\Theta(e) = \|\sqrt{\psi} e(T, \cdot)\|_{L^2(\Omega_a)}.$$

Possible choices for ψ are a Gaussian with narrow band centered at $x = K$ or $\psi = 1\!\!1_\omega$, in which case $\Theta(e) = \|e(T,\cdot)\|_{L^2(\omega)}$.

As a second example, consider a calibration problem where a volatility map is recovered from market data assimilation by minimizing some appropriate difference between model predictions and observed market prices. Assume for instance that we can observe market put option prices for three dates of maturity $t_1 < t_2 < T$ and strike prices living in an open set $\omega \subset \Omega_a$. The symmetry between call option and put option prices implies that it is equivalent to observe call option prices with fixed strike price K for initial value varying in ω. In order to recover the parameter of the model (such as volatility) from market data, we need to compute very accurately the implied option prices only in the neighborhood of the relevant strike price and dates of maturity. Hence, we may choose a control of the following L^2 norms of e

$$\Theta(e) = \|e\|_{L^2((t_1,t_2)\times\omega)} + \|e(T,\cdot)\|_{L^2(\omega)},$$

which corresponds to

$$\theta_1(t,x) = \frac{1}{||e||_{L^2((t_1,t_2)\times\omega)}}\, e(t,x)1\!\!1_{(t_1,t_2)\times\omega} \quad \text{and} \quad \theta_2(x) = \frac{1}{||e(T,\cdot)||_{L^2(\omega)}}\, e(T,x)1\!\!1_\omega.$$

We bring this short discussion on output functionals to a close by noticing that the functionals θ_1 and θ_2 must depend on e in order to control the error in some norm. As a result, they are not known a priori and in a numerical implementation, this dependence must be relaxed by means of an iterative technique. More details shall be given in Section 4. An alternative approach is to consider functionals θ_1 and θ_2 independent of e. For example, in the calibration problem, one may choose $\theta_1(t,x) = 1\!\!1_{(t_1,t_2)\times\omega}$ and $\theta_2(x) = 1\!\!1_\omega$ yielding

$$\Theta(e) = \int_{t_1}^{t_2}\int_\omega e(t,x)\,dtdx + \int_\omega e(T,x)\,dx.$$

In this case, the method controls the error in a semi-norm only. It may still be accurate for problems where the error e does not change sign.

3.2 Error representation by duality

The a posteriori error estimator with respect to the θ-error measure given by (12) is obtained by using duality arguments. The dual problem associated with (θ_1, θ_2) reads

$$\left\{\begin{array}{l} \text{find } z \in W(0,T;H_0^1(\Omega_a)) \text{ such that} \\ -\displaystyle\int_0^T \langle z_t, v\rangle_{-1,1}\,dt + \int_0^T a(v,z)\,dt = \int_0^T (\theta_1, v)_{\Omega_a}\,dt, \quad \forall v \in W(0,T;H_0^1(\Omega_a)), \\ z(T,\cdot) = \theta_2 \quad \text{in } L^2(\Omega_a). \end{array}\right. \tag{13}$$

The a posteriori error analysis will be performed under the assumptions that $\theta_1 \in H^1(0,T;H^2(\Omega_a))$ and $\theta_2 \in H^2(\Omega_a)$. In this case, L^p regularity results for evolution problems (see for instance [11, 4]) imply that the dual problem (13) has a unique solution z which belongs to $H^2(0,T;H^2(\Omega_a)\cap H_0^1(\Omega_a))$. As before, duality pairing will be remplaced by $L^2(\Omega_a)$ inner product.

Proposition 1. *The θ-error measure satisfies $\Theta(e) = B(e,z)$.*

Proof. Since by definition,

$$B(e,z) = \sum_{n=1}^{N} \int_{I_n} \{(e_t, z)_{\Omega_a} + a(e,z)\} + \sum_{n=1}^{N} ([e]_{n-1}, z_{n-1}^+)_{\Omega_a},$$

we get after integrating the component $(e_t, z)_{\Omega_a}$ by parts,

$$\begin{aligned} B(e,z) &= \sum_{n=1}^{N} \left\{ ([e]_{n-1}, z_{n-1}^+)_{\Omega_a} + [(e,z)]_{t_{n-1}}^{t_n} \right\} \\ &+ \sum_{n=1}^{N} \int_{I_n} \{(-z_t, e)_{\Omega_a} + a(e,z)\}\, dt. \end{aligned}$$

Since z solves the dual problem (13), the first term in the right-hand side of the previous equality reduces to $(\theta_2, e(T,\cdot))_{\Omega_a}$ while the second term yields $\int_0^T (\theta_1, e)_{\Omega_a}\, dt$. □

Using the consistency of the variational problem (7) and Galerkin orthogonality property, we eliminate the exact solution u from the above error representation.

Proposition 2. *Let U be the discrete solution satisfying (7), let z be the unique solution of (13) and let Z be an arbitrary test function in $P_{d/c}^{q,p}$ such that $(z - Z)_0^+ = 0$. Then, we have the error representation*

$$\Theta(e) = \sum_{n=1}^{N} \left\{ \int_{I_n} (f - U_t, z - Z)_{\Omega_a}\, dt - \int_{I_n} a(U, z - Z)\, dt - ([U]_{n-1}, (z - Z)_{n-1}^+)_{\Omega_a} \right\}. \qquad (14)$$

Proof. Since the solution u of (6) also satisfies the variational formulation (7), we have the Galerkin orthogonality property $B(e, Z) = 0$ for an arbitrary function Z in $P_{d/c}^{q,p}$. Therefore, we get

$$\begin{aligned} \Theta(e) &= B(e,z) = B(u - U, z - Z) \\ &= B(u, z - Z) - B(U, z - Z) = F(z - Z) - B(U, z - Z). \end{aligned}$$

We conclude using definitions (8) and (9). □

3.3 Localization of the error estimator

Our goal is to localize the θ-error measure given by (14) to the space-time cells $I_n \times K_i^n$. The contribution associated with each space-time cell may then be used for the purpose of refinement or derefinement as discussed in Section 4.

Let us first introduce some notation. For functions f depending on time and/or space, we denote by $||f||_{I_n \times K_i^n}$, $||f||_{K_i^n}$ and $||f||_{I_n}$ the L^2 norms taken over the corresponding subscript and we use a similar notation for L^2 scalar products. For $x \in \partial K_i^n$ and $t \in I_n$, let

$$[U_x](t) := \lim_{s \to 0^+} \{U_x(t, x_i + s) - U_x(t, x_i - s)\},$$

be the jump at x of the first derivative of U. We also introduce the computable residual $R(U) = f - \mathcal{A}U - U_t$ where $\mathcal{A}$ is the differential operator given by (4).

Proposition 3. *Keeping the assumptions of proposition 2, we have*

$$\Theta(e) = \sum_{n=1}^{N}\sum_{i=0}^{M_n} \left\{ (R(U), z-Z)_{I_n\times K_i^n} - \tfrac{1}{2}([\hat{\sigma}U_x], z-Z)_{I_n\times \partial K_i^n} - ([U]_{n-1}, (z-Z)_{n-1}^+)_{K_i^n} \right\}. \qquad (15)$$

Proof. It directly results from (14) upon integrating by parts and elementary reordering of terms. □

In order to use numerically the a posteriori estimate for the θ-error measure derived in proposition 3, we have to approximate the interpolation error $z - Z$ by higher order derivatives of the dual solution z. The numerical approximation of these derivatives is discussed in Section 4.

Proposition 4. *Assume $\theta_1 \in H^1(0,T;H^2(\Omega_a))$ and $\theta_2 \in H^2(\Omega_a)$. There exists a constant $c > 0$ independent of the space-time mesh such that: (i) For the $dG(0)$ finite element method applied to the generalized Black-Scholes problem (5), we have the a posteriori estimate*

$$|\Theta(e)| \leq c \sum_{n=1}^{N}\sum_{i=0}^{M_n} \left\{ \rho_1^{i,n}(\omega_{1,k}^{i,n} + \omega_{1,h}^{i,n}) + \rho_2^{i,n}\omega_{2,k}^{i,n} + \rho_3^{i,n}(\omega_{3,k}^{i,n} + \omega_{3,h}^{i,n}) \right\},$$

with

$$\rho_1^{i,n} = ||R(U)||_{I_n\times K_i^n}, \quad \omega_{1,k}^{i,n} = k_n||z_t||_{I_n\times K_i^n}, \quad \omega_{1,h}^{i,n} = (h_i^n)^2||z_{xx}||_{I_n\times K_i^n},$$

$$\rho_2^{i,n} = (h_i^n)^{-\frac{1}{2}}||\hat{\sigma}[U_x]||_{I_n\times \partial K_i^n}, \quad \omega_{2,k}^{i,n} = k_n\left(||z_t||_{I_n\times K_i^n} + h_i^n||z_{tx}||_{I_n\times K_i^n}\right),$$

$$\rho_3^{i,n} = k_n^{-\frac{1}{2}}||[U]_{n-1}||_{K_i^n}, \quad \omega_{3,k}^{i,n} = k_n||z_t||_{I_n\times K_i^n}, \quad \omega_{3,h}^{i,n} = (h_i^n)^2||z_{xx}||_{I_n\times K_i^n}.$$

(ii) For the $dG(1)$ finite element method applied to the generalized Black-Scholes problem (5), we have the a posteriori estimate

$$|\Theta(e)| \leq c \sum_{n=1}^{N}\sum_{i=0}^{M_n} \left\{ \rho_1^{i,n}(\omega_{1,k}^{i,n} + \omega_{1,h}^{i,n}) + \rho_2^{i,n}\omega_{2,k}^{i,n} + \rho_1^{3,n}(\omega_{3,k}^{i,n} + \omega_{3,h}^{i,n}) \right\},$$

with

$$\rho_1^{i,n} = ||R(U)||_{I_n\times K_i^n}, \quad \omega_{1,k}^{i,n} = k_n^2||z_{tt}||_{I_n\times K_i^n}, \quad \omega_{1,h}^{i,n} = (h_i^n)^2||z_{xx}||_{I_n\times K_i^n},$$

$$\rho_2^{i,n} = (h_i^n)^{-\frac{1}{2}}||\hat{\sigma}[U_x]||_{I_n\times \partial K_i^n}, \quad \omega_{2,k}^{i,n} = k_n^2\left(||z_{tt}||_{I_n\times K_i^n} + h_i^n||z_{ttx}||_{I_n\times K_i^n}\right),$$

$$\rho_3^{i,n} = k_n^{-\frac{1}{2}}||[U]_{n-1}||_{K_i^n}, \quad \omega_{3,k}^{i,n} = k_n^2||z_{tt}||_{I_n\times K_i^n}, \quad \omega_{3,h}^{i,n} = (h_i^n)^2\{||z_{xx}(t_{n-1})||_{I_n\times K_i^n} + k_n||z_{txx}||_{I_n\times K_i^n}\}.$$

Remark 1. *The quantities ρ are residuals depending on the finite element solution U and the quantities ω are weights depending on the dual solution z. The dual solution brings information where the numerical error is generated. Thus, it enables to account for error propagation in space and time. For instance, a small residual but a large weight at a given cell $I_n \times K_i^n$ indicate that this cell may actually contribute significantly to error generation.*

Remark 2. *The assumptions on the error control functions θ_1 and θ_2 ensure that all the weights are well defined. From a theoretical viewpoint, these assumptions may be slightly relaxed. From a practical viewpoint, it is interesting to consider error control functions that are proportional to the error $e = u - U$. In this case, the best regularity for θ_1 and θ_2 can only be $\theta_1 \in L^2(0,T;H^1(\Omega_a))$ and $\theta_2 \in H^1(\Omega_a)$. This approach will be illustrated numerically in Section 5.*

Proof. The proof directly follows the techniques derived in [5, 8]. For completeness, we detail the proof in the dG(0) case and only sketch it in the dG(1) case.
(i) The key point is the choice of the interpolant Z in $P^{0,1}_{d/c}$. For a function $w \in L^2(S_n)$, its orthogonal L^2-projection on the subspace spanned by the functions that are constant in time is given by

$$(P_n w)(x) = \frac{1}{k_n} \int_{I_n} w(s,x)\, ds.$$

Moreover, for a function $w \in C^0(\Omega_a)$, its Lagrange interpolant on mesh $\mathcal{M}_n$ is given by

$$(\tilde{I}_n w)(x) = \sum_{i=1}^{M_n} w(x_i)\phi_i^n(x).$$

Following Eriksson and Johnson [5], we take $Z(t,x) = \sum_{n=1}^{N} (\tilde{I}_n \circ P_n z)(x) \mathtt{1}_{\{t \in I_n\}}$, i.e. for $(t,x) \in S_n$, we have

$$Z(t,x) = \sum_{i=1}^{M_n} \left(\frac{1}{k_n} \int_{I_n} z(s,x_i)\, ds \right) \phi_i^n(x).$$

We write $z - Z = z - P_n z + (id - \tilde{I}_n) \circ P_n z$ so that the control of any norm $||z - Z||$ will be achieved by a control of both the time contribution $||z - P_n z||$ and the space contribution $||(id - \tilde{I}_n) \circ P_n z||$.
(ii) Consider the time contribution. For $(t,x) \in S_n$, we have

$$z(t,x) = z(t_{n-1},x) + \int_{t_{n-1}}^{t} z_t(s,x)\, ds.$$

Therefore, for every $x \in \Omega_a$, the characteristic property of the L^2-projection yields

$$||z - P_n z||_{I_n} \leq ||z - z(t_{n-1},.)||_{I_n} = \left(\int_{I_n} \left(\int_{I_n} |z_t(s,x)|\, ds \right)^2 \right)^{\frac{1}{2}} \leq k_n ||z_t||_{I_n}. \tag{16}$$

Thus, the quantities $\rho_1^{i,n}$ and $\omega_{1,k}^{i,n}$ result from the estimates

$$\begin{aligned} (R(U), z - P_n z)_{I_n \times K_i^n} &\leq \int_{x_i^n}^{x_{i+1}^n} ||R(U)||_{I_n} ||z - P_n z||_{I_n}\, dx \\ &\leq k_n \int_{x_i^n}^{x_{i+1}^n} ||R(U)||_{I_n} ||z_t||_{I_n}\, dx \leq \underbrace{||R(U)||_{I_n \times K_i^n}}_{\rho_1^{i,n}} \underbrace{k_n\, ||z_t||_{I_n \times K_i^n}}_{\omega_{1,k}^{i,n}}. \end{aligned}$$

Next, owing to the standard trace inequality (for a proof, see [14])

$$\forall w \in H^1(K_i^n), \qquad ||w||_{\partial K_i^n} \leq c_1 \left\{ \frac{1}{\sqrt{h_i^n}} ||w||_{K_i^n} + \sqrt{h_i^n} ||w_x||_{K_i^n} \right\}$$

where c_1 is independent of K_i^n and w, we deduce that for every $w \in L^2(I_n, H^1(K_i^n))$ holds

$$||w||_{I_n \times \partial K_i^n} = \left(\int_{I_n} ||w||^2_{\partial K_i^n}\, dt \right)^{\frac{1}{2}} \leq c_1 \left\{ \frac{1}{\sqrt{h_i^n}} ||w||_{I_n \times K_i^n} + \sqrt{h_i^n} ||w_x||_{I_n \times K_i^n} \right\}. \tag{17}$$

Since $z_t \in L^2(I_n, H^1(K_i^n))$, we obtain using (16) and (17) that

$$\begin{aligned}
\tfrac{1}{2}([\hat{\sigma}U_x], z - P_n z)_{I_n \times \partial K_i^n} &\leq \tfrac{1}{2} |||[\hat{\sigma}U_x]|||_{I_n \times \partial K_i^n} \left(\int_{\partial K_i^n} ||z - P_n z||_{I_n}^2 \, dx \right)^{1/2} \\
&\leq \tfrac{1}{2} |||[\hat{\sigma}U_x]|||_{I_n \times \partial K_i^n} \, k_n ||z_t||_{I_n \times \partial K_i^n} \\
&\leq \tfrac{c_1}{2} \underbrace{(h_i^n)^{-1/2} |||[\hat{\sigma}U_x]|||_{I_n \times \partial K_i^n}}_{\rho_2^{i,n}} \underbrace{k_n \left(||z_t||_{I_n \times K_i^n} + h_i^n ||z_{tx}||_{I_n \times K_i^n} \right)}_{\omega_{2,k}^{i,n}}.
\end{aligned}$$

Finally, for every $x \in K_i^n$, integration by parts in time yields

$$\begin{aligned}
(z - P_n z)_{n-1}^+(x) &= (z - P_n z)(t_{n-1}, x) = -\frac{1}{k_n} \int_{I_n} (t_n - s)(z - P_n z)_t(s, x) \, ds \\
&= -\frac{1}{k_n} \int_{I_n} (t_n - s) z_t(s, x) \, ds \leq \frac{1}{\sqrt{3}} \sqrt{k_n} ||z_t||_{I_n}.
\end{aligned}$$

Therefore,

$$([U]_{n-1}, (z - P_n z)_{n-1}^+)_{K_i^n} \leq \tfrac{1}{\sqrt{3}} \underbrace{k_n^{-\frac{1}{2}} |||[U]_{n-1}||_{K_i^n}}_{\rho_3^{i,n}} \underbrace{k_n ||z_t||_{I_n \times K_i^n}}_{\omega_{3,k}^{i,n}}.$$

(iii) Consider now the space contribution. Recalling the standard finite element interpolation estimate valid in 1D

$$\forall w \in H^2(\Omega_a) \cap H_0^1(\Omega_a), \qquad ||(id - \tilde{I}_n) w||_{K_i^n} \leq c_2 \, (h_i^n)^2 \, ||w_{xx}||_{K_i^n},$$

with $c_2 = \frac{1}{\pi^2}$ and keeping in mind that $P_n z$ belongs to $H^2(\Omega_a) \cap H_0^1(\Omega_a)$ with

$$\begin{aligned}
||(P_n z)_{xx}||_{I_n \times K_i^n} &= ||P_n(z_{xx})||_{I_n \times K_i^n} = \left(\int_{K_i^n} ||P_n(z_{xx})||_{I_n}^2 \, dx \right)^{\frac{1}{2}} \\
&\leq \left(\int_{K_i^n} ||z_{xx}||_{I_n}^2 \, dx \right)^{\frac{1}{2}} = ||z_{xx}||_{I_n \times K_i^n},
\end{aligned}$$

we get

$$\begin{aligned}
(R(U), (id - \tilde{I}_n) \circ P_n z)_{I_n \times K_i^n} &\leq \int_{I_n} ||R(U)||_{K_i^n} \, ||(id - \tilde{I}_n) \circ P_n z||_{K_i^n} \, dt \\
&\leq c_2 \underbrace{||R(U)||_{I_n \times K_i^n}}_{\rho_1^{i,n}} \underbrace{(h_i^n)^2 ||z_{xx}||_{I_n \times K_i^n}}_{\omega_{1,h}^{i,n}}.
\end{aligned}$$

We notice next that in one space dimension, $P_n z - Z = 0$ on $\partial K_i^n \times I_n$ by construction. Therefore, $([\hat{\sigma}U_x], P_n z - Z)_{I_n \times \partial K_i^n} = 0$ and the weight $\omega_{2,h}^{i,n}$ vanishes. For higher space dimension, this is no longer the case (see [8] for the value of $\omega_{2,h}^{i,n}$ in the case of the heat equation). Finally, we have

$$\begin{aligned}
([U]_{n-1}, ((id - \tilde{I}_n) \circ P_n z)_{n-1}^+)_{K_i^n} &\leq ||[U]_{n-1}||_{K_i^n} \, ||((id - \tilde{I}_n) \circ P_n z)_{n-1}^+||_{K_i^n} \\
&= (k_n)^{-\frac{1}{2}} |||[U]_{n-1}||_{K_i^n} \, ||((id - \tilde{I}_n) \circ P_n z)_{n-1}^+||_{I_n \times K_i^n} \\
&\leq c_2 \underbrace{(k_n)^{-\frac{1}{2}} |||[U]_{n-1}||_{K_i^n}}_{\rho_3^{i,n}} \underbrace{(h_i^n)^2 ||z_{xx}||_{I_n \times K_i^n}}_{\omega_{3,h}^{i,n}}.
\end{aligned}$$

This completes the proof in the dG(0) case.
(iv) In the dG(1) case, P_n is the orthogonal L^2-projection on the subspace spanned by the functions that are affine in time. We then have the estimate $||z - P_n z||_{I_n} \leq k_n^2 ||z_{tt}||_{I_n}$. Two integrations by parts in time are performed to estimate $\omega_{3,k}^{i,n}$. Finally, when estimating $\omega_{3,h}^{i,n}$, a trace inequality in time is used in order to control the norm over K_i^n by norms over $I_n \times K_i^n$. □

Remark 3. *When σ is constant in time, we have $([\hat{\sigma} U_x], z - P_n z)_{I_n \times \partial K_i^n} = 0$ by definition of the orthogonal L^2-projection operator. Therefore, we may take $\rho_2^{i,n} = 0$. This remark applies to both dG(0) and dG(1) cases.*

Remark 4. *The exact value of the interpolation constant c arising in the estimates of proposition 4 is given in the proof above and one generally has $c \leq 1$. In our numerical experiments, we will simply take $c = 1$.*

4 Adaptive mesh refinement

In this section we present an adaptive mesh refinement/derefinement algorithm based on the a posteriori error estimate derived in the previous section. Particular emphasis is laid upon practical implementation aspects.

4.1 The algorithm

Given a tolerance `tol`, our goal is to contruct adaptively a computational mesh on which the discrete solution U achieves the accuracy requirement

$$|\Theta(u - U)| \leq \texttt{tol}.$$

To this purpose, we first notice that the a posteriori error estimator obtained in proposition 4 allows to separate the contribution due to space and time discretization. More precisely, for a time slab S_n, let us define

$$\eta_k^n = \sum_{i=0}^{M_n} \eta_k^{i,n} \quad \text{where} \quad \eta_k^{i,n} = \sum_{j=1}^{3} \rho_j^{i,n} \omega_{j,k}^{i,n},$$

and

$$\eta_h^n = \sum_{i=0}^{M_n} \eta_h^{i,n} \quad \text{where} \quad \eta_h^{i,n} = \rho_1^{i,n} \omega_{1,h}^{i,n} + \rho_3^{i,n} \omega_{3,h}^{i,n}.$$

The global time and space contribution are respectively defined as

$$\eta_k = \sum_{n=1}^{N} \eta_k^n, \qquad \eta_h = \sum_{n=1}^{N} \eta_h^n.$$

Error control will be achieved by imposing that both η_h and η_k be lower than $\frac{\texttt{tol}}{2}$. Other choices modifying the balance between space and time contributions might be considered as well.

The iterative algorithm by which the space-time mesh is adaptively modified reads as follows. We use the notation $\Gamma = \cup_{n=1}^{N} I_n \times \mathcal{M}_n$ for a given space-time mesh and denote by Γ_i the mesh at iteration i of the adaptive algorithm. In addition, the parameter ε denotes a derefinement threshold with $0 < \varepsilon < 1$.

1. Construct an initial space-time mesh Γ_0. Γ_0 may typically be a rather coarse, uniform, tensor-product mesh. Set $j = 0$;

2. Compute discrete solution U_j of primal problem (5) on space-time mesh Γ_j. Compute residuals $\rho_j^{i,n}$ for $j = 1, 2, 3$;

3. Compute error $e_j = u - U_j$ and evaluate error control functions θ_1 and θ_2;

4. Solve dual problem (13) and compute weights $\omega_{j,k}^{i,n}$ and $\omega_{j,h}^{i,n}$ for $j = 1, 2, 3$;

5. Compute error estimators η_h^n and η_k^n. If $|\Theta(e_j)| \leq \texttt{tol}$ then `STOP`, else construct a new mesh Γ_{j+1} using the following refinement/derefinement procedure:

   5.a Space loop: let $M = \sum_{n=1}^{N} M_n$ be the total numbers of elements. If $\eta_h^{i,n} \geq \frac{\texttt{tol}}{2M}$, we refine by halving the cell K_i^n while if $\eta_h^{i,n} \leq (1-\varepsilon)\frac{\texttt{tol}}{2M}$ for two consecutive indices i and $i+1$, we derefine by assembling the cells K_i^n and K_{i+1}^n;

   5.b Time loop: if $\eta_k^n \geq \frac{\texttt{tol}}{2N}$, we refine by halving the time interval I_n while if $\eta_k^n \leq (1-\varepsilon)\frac{\texttt{tol}}{2N}$ for two consecutive indices n and $n+1$, we derefine by assembling the time intervals I_n and I_{n+1};

   Set $j := j + 1$ and return to step 2.

The above algorithm thus consists in a sequence of forward/backward sweeps where a primal problem and a dual problem are sequentially solved on a given space-time mesh which is then adaptively modified. An alternative approach developed for instance in [10] is based on L^∞ estimates in time and cubic Hermite polynomial interpolation in space and allows to implement a single forward time-marching algorithm where slabs are iteratively refined one after the other.

4.2 Practical implementation

The adaptive algorithm described in the previous section needs several modifications to be useful in practice.

4.2.1 Approximate data for dual problem

For European options with constant volatility, closed formulas are available to evaluate the exact solution u and therefore the error $e_j = u - U_j$. However, in the financially interesting case where local volatility is considered, closed formulas are no longer available and the data for the dual problem needs to be estimated.

A simple procedure is to use a relaxation method in which $e \simeq U_j - U_{j-1}$. This approximation may be partly justified under a saturation assumption of the form

$$\|e_j\| \leq \beta \, \|e_{j-1}\|,$$

for some $0 < \beta < 1$ independent of j. This assumption yields

$$\tfrac{1-\beta}{\beta}\|e_j\| \leq \|U_j - U_{j-1}\| \leq (1+\beta)\|e_{j-1}\|,$$

showing that the estimate $\|U_j - U_{j-1}\|$ is asymptotically equivalent to the error.

With this modification, the first three steps of the adaptive algorithm are modified as follows:

1. Construct an initial space-time mesh Γ_0 and a second space-time mesh Γ_1 by global refinement of Γ_0 in space and time. Compute discrete solution U_0 of primal problem (5) on initial space-time mesh Γ_0. Set $j = 1$;

2. Compute discrete solution U_j of primal problem (5) on space-time mesh Γ_j. Compute residuals $\rho_j^{i,n}$ for $j = 1, 2, 3$;

3. Approximate error $e_j \simeq U_j - U_{j-1}$ and evaluate error control functions θ_1 and θ_2.

4.2.2 Discrete dual problem

Several approaches have been investigated in the literature to solve approximately the dual problem. One of such approaches consists in using higher-order interpolation to estimate the dual solution z (see for instance [12] and [2]). This approach usually yields sharp bounds for the θ-error measure but is not straightforward to implement since it generally needs some restrictions on the used meshes. For instance, it is convenient to utilize meshes that are organized patch-wise with local hierarchical refinement. An alternative approach, which yields looser bounds for the θ-error measure but is easier to implement, is to discretize the dual problem on the same mesh and with the same polynomial interpolation as the primal problem. This second approach has been selected hereafter. Our numerical experiments show that for the finance problems under consideration, this choice yields nearly optimal convergence rates.

The discrete dual problem reads

$$\begin{cases} \text{find } Z \in P_{d/c}^{q,p} \text{ such that} \\ B(V, Z) = \Theta(V), \quad \forall V \in P_{d/c}^{q,p}, \end{cases} \tag{18}$$

with Θ defined in (12).

4.2.3 Computing the weights

Whenever possible, integrals are evaluated analytically or are approximated by 4-point Gaussian quadrature. In order to estimate the weights involving high order derivatives (z_{xx}, z_{tt}, z_{txx} and z_{ttx}) from the discrete dual solution Z, we use the following approximations:

- the second order derivative in space is approximated by

$$\left(\frac{1}{2}((Z_{xx}^2(t_n, x_i) + Z_{xx}^2(t_n, x_{i-1})) \times \text{mes}(I_n \times k_i^n)\right)^{\frac{1}{2}},$$

  with $Z_{xx}(t_n, x_i)$ computed from

$$\frac{2}{h_i + h_{i+1}} \left(\frac{Z(t_n, x_{i+1}) - Z(t_n, x_i)}{h_{i+1}} - \frac{Z(t_n, x_i) - Z(t_n, x_{i-1})}{h_i} \right).$$

- the second order derivative in time is approximated by

$$\left(\frac{1}{2}(Z_{tt}^2(t_n, x_i) + Z_{tt}^2(t_n, x_{i-1})) \times \text{mes}(I_n \times K_i^n)\right)^{\frac{1}{2}},$$

with $Z_{tt}(t_n, x_i)$ computed from

$$\frac{2}{k_n + k_{n+1}} \left(P_n Z_t(t_{n+1}, x_i) - Z_t(t_n, x_i) \right),$$

and $Z_t(t_n, x_i)$ is directly recovered from the local, linear-in-time expression for Z.

- for the third order derivatives z_{txx} and z_{ttx}, the previous quadratures are used with Z replaced by Z_t and Z_x respectively.

5 Numerical Results

In this section we illustrate numerically the adaptive finite element method on two test problems. We first study a standard Black-Scholes model with constant volatility in order to assess the numerical behavior of our algorithms. As a more realistic model for finance applications, we then investigate a Black-Scholes model with local volatility.

All the computations have been performed in double precision on a PC Athlon 1.2 GHz with 256 Mb of RAM. In the dG(0) case, linear systems were solved using Crout factorization whereas in the dG(1) case, SOR algorithm was employed.

5.1 Standard Black-Scholes Model

We consider a standard Black-Scholes model for pricing European call options with payoff $\psi = (S - K)_+$, exercise price $K = 100$, volatility $\sigma = 0.2$, maturity $T = 1$ year, instantaneous interest rate $r = \log(1.1)$ and dividend yield rate $\delta = 0$. We compute the solution on the localized domain $[10, 1000]$.

We first verify numerically the convergence rate of the dG(1) method. Numerical results obtained by simply doubling both the number of time steps and spatial cells from one mesh to the next finer one are reported in table 1. N_{ad} is the index for adaptive mesh iteration, N_{cell} the total number of space-time cells, N the number of time steps, η_h and η_k the space and time contribution to the a posteriori error indicator η, $I_{\text{eff}} = \eta / ||e_N^-||_{\Omega_a}$ the efficiency index and CPU the computation time. For standard Black-Scholes models, CPU times will be normalized by the one corresponding to the initial mesh for the first test case described in table 1.

N_{ad}	N_{cell}	N	η_h	η_k	η	order	$\|\|e_N^-\|\|_{\Omega_a}$	order	I_{eff}	CPU
0	1024	16	10.29	9.9e-04	10.29	—	2.72	—	3.77	1
1	4096	32	3.38	6.7e-05	3.38	1.60	0.92	1.56	3.64	13
2	16384	64	0.93	5.0e-06	0.93	1.75	0.20	2.20	4.46	213
3	65536	128	0.22	1.0e-07	0.22	2.07	6.1e-0.2	1.71	3.73	1077
4	262144	256	0.06	1.0e-07	0.06	1.85	8.3e-0.3	2.87	7.44	4686
5	1048576	512	0.02	1.0e-07	0.02	1.88	2.7e-0.3	1.62	6.51	5445

Table 1: Standard Black-Scholes model, dG(1) method; meshes are obtained by simply doubling both the number of time steps and spatial cells from one mesh to the next finer one.

On all meshes, the spatial error dominates the time error. Both the a posteriori error bound η and the actual error $||e_N^-||_{\Omega_a}$ are theoretically second order in space and time. In practice, we observe convergence orders fluctuating between 1.6 and 2.1. The efficiency index takes values between 3 and

8 approximately. The fact that $I_{\text{eff}} \geq 1$ confirms the reliability of the method, i.e. η less than a given tolerance actually implies that $||e_N^-||_{\Omega_a}$ lies below the same threshold. Values of the efficiency index closer to 1 can be achieved by incorporating an appropriate interpolation constant in the error indicator η. Indeed, in our calculations, we have simply set this constant to one, whereas common numerical practice suggests taking values ranging from 0.1 to 0.5 (see for instance [8]).

We next control the error $||e_N^-||_{\Omega_a}$ at maturity using the adaptive mesh strategy. To this purpose, we choose the control functions $\theta_1 = 0$ and $\theta_2(x) = e_N^-(x)/||e_N^-||_{\Omega_a}$. We take $\texttt{tol} = 0.002$ and $\varepsilon = 0.2$. Numerical results are presented in table 2. After 6 steps, the error $||e_N^-||_{\Omega_a}$ has been brought below the prescribed tolerance. The fifth mesh yields the same accuracy as the uniformly refined mesh but contains approximately 100 times less space-time cells, resulting in substantial CPU savings. The efficiency I_{eff} again ranges between 3 and 7 confirming the reliability of the method and the fact that looser interpolation constants may be used. Although the function θ_2 is not smooth enough to satisfy the assumptions of Proposition 4, the above numerical results confirm that it may be used in practice. Regularized versions of θ_2 might be considered as well but would not yield any significant improvement from a financial viewpoint.

N_{ad}	N_{cell}	N	η_h	η_k	η	$\|\|e_N^-\|\|_{\Omega_a}$	I_{eff}	CPU
0	1024	16	10.29	9.9e-04	10.29	2.72	3.77	1.0
1	1155	21	3.11	1.4e-03	3.12	0.92	3.36	1.6
2	1325	17	0.85	9.8e-04	0.85	0.20	4.10	3.1
3	2309	21	0.20	3.4e-03	0.21	6.1-02	3.45	7.3
4	3339	18	5.8e-02	1.7e-03	5.9e-02	8.5e-03	6.98	17.6
5	10177	31	1.3e-02	5.9e-04	1.4e-02	3.0e-03	4.55	62.7
6	12989	28	3.8e-03	2.0e-03	5.8e-03	1.9e-03	3.01	168.5

Table 2: Standard Black-Scholes model, dG(1) method; adaptive mesh refinement designed to control the error at maturity over the whole price range.

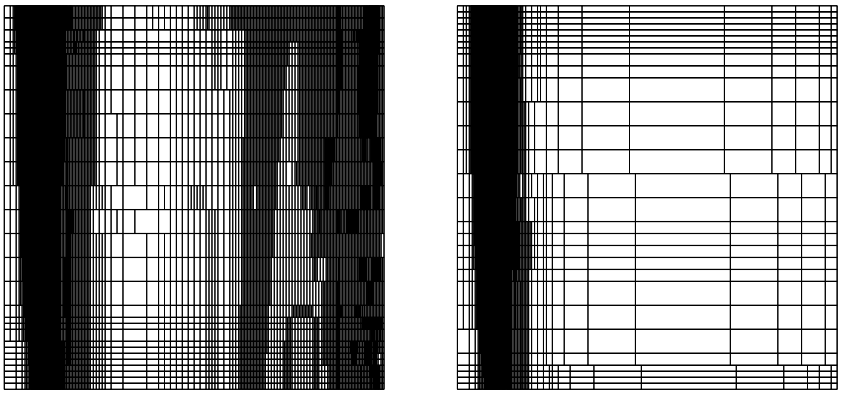

Figure 2: Standard Black-Scholes model, dG(1) method; final mesh controling the error over whole interval Ω_a (left) and only at vicinity of exercise price (right).

Figure 2 (left) presents the final mesh that was adaptively generated in order to reach the prescribed accuracy goal. The finest space-time cells concentrate in two zones of the computational domain: in a region situated around the exercise price K and also near the boundary corresponding to high

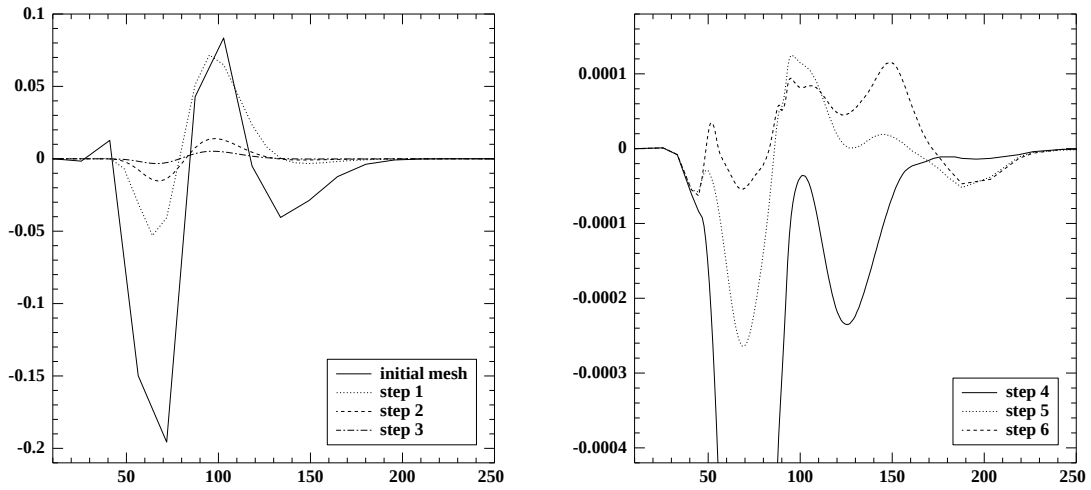

Figure 3: Standard Black-Scholes model, dG(1) method; spatial distribution of the error at maturity for the seven meshes considered in table 2; left: 4 coarser ones; right: 3 finer ones.

values of the dividend paying stock S_t. While the first region is certainly desirable for accurate option pricing, the second arises from the fact that by controlling the maturity error over the whole interval Ω_a, a significant part of the computational effort is devoted to the reduction of the error for high values of S_t. Figure 3 presents the spatial distribution of the error at maturity obtained on the seven meshes considered in table 2. Although error control is achieved in the $L^2(\Omega_a)$ norm, we also observe convergence in the L^∞ norm in the vicinity of the exercise price.

In order to avoid unnecessary refinements in the vicinity of the right border of Ω_a, we consider the control function

$$\theta_2(x) = \frac{e_N^-(x)}{||e_N^-||_{[50,150]}} 1\!\!1_{\{x\in[50,150]\}},$$

yielding a posteriori control of the error at maturity in the vicinity of the exercise price. Numerical results are reported in table 3. Figure 2 (right) presents the finest mesh considered in table 3. Mesh refinement only occurs in the vicinity of the exercise price, resulting in additional CPU savings.

N_{ad}	N_{cell}	N	η_h	η_k	η	$\|\|e_N^-\|\|_{[50,150]}$	I_{eff}	CPU
0	1024	16	10.29	1.0e-03	10.29	2.72	3.77	1.0
1	1266	23	3.18	1.0e-03	3.18	0.92	3.43	2.2
2	995	17	0.85	1.2e-04	0.85	0.20	4.10	3.5
3	1607	20	0.20	1.5e-03	0.20	6.1e-02	3.40	11.1
4	2330	19	5.8e-02	1.7e-03	5.9e-02	8.3e-03	7.20	25.0
5	6488	30	1.4e-02	9.0e-04	1.5e-02	2.7e-03	5.56	67.4
6	9994	29	5.7e-03	4.3e-03	1.0e-02	1.1e-03	8.87	151.8

Table 3: Standard Black-Scholes model, dG(1) method; adaptive mesh refinement designed to control the error at maturity in the vicinity of the exercise price.

Because of the simplicity of the standard Black-Scholes model, all the preceding results have been

obtained with the data for the dual problem evaluated from a closed formula for the exact solution. When considering more complex models, closed formulas are no longer available and the data for the dual problem needs to be estimated using the relaxation procedure outlined in Section 4.2.1. Numerical results are presented in table 4. Overall accuracy degradation remains marginal. Indeed, the approximate dual problem still provides enough information to decrease the error significantly and to maintain reliability for the error indicator.

N_{ad}	N_{cell}	N	η_h	η_k	η	$\|\|e_N^-\|\|_{\Omega_a}$	I_{eff}	CPU
0	1024	16	3.31	1.1e-03	3.32	2.72	1.21	1.1
1	1398	23	5.38	1.3e-03	5.39	0.92	5.81	2.7
2	3133	38	1.87	1.6e-03	1.88	0.20	9.00	7.2
3	3029	27	0.51	9.5e-04	0.51	6.1e-02	8.57	2.5
4	4646	27	0.10	1.9e-03	0.10	8.5e-03	12.71	11.5
5	7588	28	1.8e-02	6.9e-04	1.9e-02	3.0e-03	6.12	57.2
6	10381	24	8.2e-03	2.2e-03	1.0e-02	2.4e-03	4.26	140.9

Table 4: Standard Black-Scholes model, dG(1) method; adaptive mesh refinement designed to control the error at maturity over whole price range; approximate data for dual problem.

To conclude the numerical experiments with the standard Black-Scholes model, we compare the dG(0) and dG(1) methods. For both methods, we control the error at maturity over the whole price range. Data for the dual problems is evaluated analytically. Numerical results for the dG(0) method are reported in table 5 and should be compared to those of table 2. The superiority of the dG(1) method appears clearly in terms of both accuracy and computational efficiency.

N_{ad}	N_{cell}	N	η_h	η_k	η	$\|\|e_N^-\|\|_{\Omega_a}$	I_{eff}	CPU
0	1024	16	13.30	2.33	15.64	2.70	5.78	0.4
1	4076	32	4.22	1.68	5.91	0.92	6.41	1.4
2	15338	64	0.85	1.22	2.07	0.25	8.16	5.4
3	59248	128	0.37	0.79	1.16	0.10	11.33	22.5
4	227206	256	0.15	0.42	0.57	5.0e-02	11.17	95.0
5	842740	512	4.8e-02	0.21	0.26	2.0e-02	10.28	459.6

Table 5: Standard Black-Scholes model, dG(0) method; adaptive mesh refinement designed to control the error at maturity over the whole price range.

5.2 Local Volatility Models

This section reports numerical results obtained with the adaptive dG(1) finite element method for pricing European call options. Option parameters are taken as in the previous section except for the volatility which is now a deterministic function of t and x

$$\sigma(t,x) = \frac{15}{x}1_{\{t\in[0,0.25)\}} + \frac{40}{x}1_{\{t\in[0.25,0.5)\}} + \frac{20}{x}1_{\{t\in(0.5,1.0]\}}.$$

We are interested in computing the solution very accurately over the price interval $[90, 110]$ and for the discrete times $t = 0.25$, 0.5 and 1.0. To achieve this goal using the adaptive procedure, we take

for the error control functions

$$\theta_1(t,x) = \frac{e(t,x)}{||e||_{[0.2,0.6]\times[80,120]}}\,1_{\{(t,x)\in[0.2,0.6]\times[80,120]\}},$$

and

$$\theta_2(x) = \frac{e_N^-(x)}{||e_N^-||_{[80,120]}}\,1_{\{x\in[80,120]\}},$$

yielding a posteriori control of the error at maturity and in the time interval $[0.2, 0.6]$ and for prices lying in the interval $[80, 120]$. Sharper bracketing of the targeted price and time intervals might be considered as well.

N_{ad}	N_{cell}	N	η_h	η_k	η	CPU
0	2048	32	11.66	0.30	11.88	1.0
1	3814	64	15.22	0.47	15.69	2.5
2	9268	128	4.70	0.10	4.81	6.2
3	17904	168	2.52	0.05	2.58	17.0
4	54031	289	0.91	0.01	0.93	10.9
5	159504	478	0.32	6.9e-03	0.33	60.3
6	508646	834	0.11	2.3e-03	0.11	924.4
7	1464810	1306	3.8e-02	8.1e-04	3.8e-02	4195.6

Table 6: Local volatility Black-Scholes model, dG(1) method; adaptive mesh refinement designed to control the error in the vicinity of the exercise price and calibration dates.

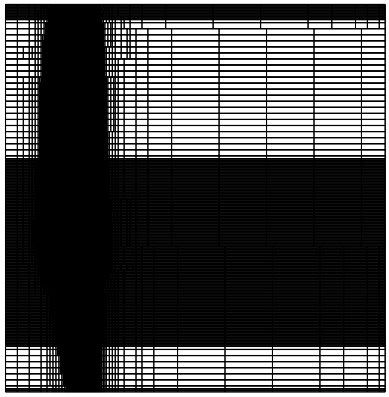

Figure 4: Local volatility Black-Scholes model, dG(1) method; third, adaptively generated mesh designed to control the error in the vicinity of the exercise price and calibration dates.

Numerical results are presented in table 6. The computational domain is set to $[10, 500]$ and the tolerance for the adaptive algorithm to $\texttt{tol} = 0.001$. CPU times are normalized by the one corresponding to the initial mesh. After seven adaptive mesh refinements, the error indicator yields $\eta = 0.038$. Based on the estimates for the interpolation constants derived for the standard Black-Scholes model, we infer that this value is reasonably compatible with the preset accuracy threshold.

Figure 4 presents the third adaptively generated mesh, showing that the finest computational cells are concentrated in the regions of financial interet for calibration purposes.

Because our space-time meshes are constructed from spatially refined time slabs, the present methodology incurs an additional cost due to excessively refined cells for high price ranges in the time interval for calibration. However, it is worthwhile to point out that if a naive procedure (doubling the number of time and spatial steps at each mesh refinement) was used instead of the adaptive algorithm, the fifth mesh would yield the same accuracy ($\eta = 0.34$) but would require 15 times the amount of CPU.

6 Conclusions

In this paper, we have investigated a relatively novel numerical method, adaptive space-time finite elements, to approximate PDEs arising in finance applications. The main appeal of the proposed methodology is that it allows the user to specify an error control function targeting its own accuracy requirements. This may be computationally effective for option pricing as well as for calibration problems where numerical errors must only be controlled in some specific areas of the whole computational domain. The method is also reliable since sharp estimates based on a posteriori estimates are available to decide whether the computational error satisfies acceptable accuracy requirements for finance applications. Compared with simple finite difference or finite element schemes based on hand taylored mesh refinement, the present methodology delivers similar or better CPU times, but offers the advantage of reliability through a posteriori error control. With current computer technology, overall wall clock times for option pricing are of the order of a few minutes.

References

[1] Barles, G., Daher C. and Romano M. (1995) Convergence of numerical schemes for problems arising in finance theory, *Math. Models and Meth. in Applied Science* 5, 125-143.

[2] Becker, R. and Rannacher, R. (2001) An optimal control approach to a posteriori error estimation in finite element methods, *Acta Numerica* 10, 1-102.

[3] Berestycki, H., Busca J. and Florent I. (2002) Asymptotics and calibration of local volatility models, *Quantitative Finance* 2, 61-69.

[4] Coulhon, T. and Lamberton, D. (1987) Régularité L^p pour les équations d'évolution, *Publications mathématiques de l'université Paris VII* 26, 155-165.

[5] Eriksson, K. and Johnson C. (1991): Adaptive finite element methods for parabolic problem: A linear model problem, *SIAM Journal of Numerical Analysis* 28, 43-77.

[6] Eriksson, K., Estep, D., Hansbo, P. and Johnson C. (1995) Introduction to adaptive methods for differential equations. *In: Iserles, A. (ed.), Acta numerica*, 105-158.

[7] Ern, A. and Guermond, J.-L. (2002) *Eléments finis : théorie, applications, mise en œuvre*, Collection Mathématiques & Applications, Vol 36, Springer, Heidelberg.

[8] Hartmann, R. (1998) A posteriori Fehlerschätzung und adaptive Schrittweiten und Ortsgittersteuerung bei Galerkin-Verfahren für die Wärmelleitungsgleichung. Diplomarbeit, Universität Heidelberg.

[9] Johnson, C. (1987) *Numerical Solution of Partial Differential Equations by the Finite Element Method*, Cambridge University Press, Cambridge.

[10] Jackson, N. and Süli, E. (1997) Adaptive finite element solutions of 1D European option pricing problems, *Technical report NA/97-05, Oxford University Computing Laboratory.*

[11] Pazy, A. (1983) *Semigroups of linear operators and applications to partial differential equations*, Applied Mathematical Sciences, Springer-Verlag, New York.

[12] Süli, E. and Houston, P. (2002) Adaptive Finite Element Approximation of Hyperbolic Problems; in *T. Barth and H. Deconinck, editors: Error Estimation and Adaptive Discretization Methods in Computational Fluid Dynamics. Lecture Notes in Computational Science and Engineering* 25, Springer-Verlag, 269-344

[13] Verfürth, R. (1996) *A review of a posteriori error estimation and adaptive mesh-refinement techniques*, Wiley, Chichester.

[14] Verfürth, R. (1998) A posteriori error estimators for convection-diffusion equations, *Numer. Math.* 80, 641-663.