Finite Element Methods for Option Pricing

Yves Achdou & Olivier Pironneau
LJLL, Université Pierre et Marie Curie, Place Jussieu, Paris F-75005
E-mail: pironneau@ann.jussieu.fr

Abstract

The finite element method is well suited to the numerical solution of the partial differential equations arising in finance because they allow for a posteriori error estimates and mesh adaptivity. The method will be described on three simple examples and its advantages will be emphasized.

Introduction

The Finite Element Method (FEM) has been invented by engineers around 1950 for solving the partial differential equations arising in solid mechanics; the main idea was to use the principle of virtual work for designing discrete approximations of the boundary value problems. The most popular reference on FEM in solid mechanics is the book of Zienckiewicz [5]. Generalizations to other fields of physics or engineering have been done by applied mathematicians through the concept of variational formulations and weak forms of the partial differential equations (see Ciarlet[6] for example).

In finance, partial differential equations (PDE) or partial integro-differential equations (PIDE) may be used for option pricing. For approximating their solutions, at least four classes of numerical methods can be used:

- **Finite Difference Methods** are by far the simplest, except when mesh adaptivity is required in which case it is rather difficult to control the numerical error.

- **Finite Volume Methods** are not really natural, because these methods are better suited for hyperbolic PDEs. In finance, they may be useful, for example for Asian options when the PDE becomes close to hyperbolic near maturity.
• **Spectral Methods** are Galerkin methods with Fourier series of high degree polynomials. They are ideal when the coefficients of the PDE are constant, which is rarely the case in financial engineering. For a very efficient adaptation of spectral methods to finance see Oosterlee et al\[3\].

• **Finite Element Methods** seem at first glance unnecessarily complex for finance where a large class of problems are one dimensional in space. Yet, they are very flexible for mesh adaptivity and the implementation difficulties are only apparent.

### 1 A Simple Example

Take the simplest case, the Black-Scholes model for a put option with a strike $K$ and a maturity $T$. The price of the underlying asset is assumed to be a geometric Brownian motion

$$dS_t = S_t(\mu dt + \sigma dW_t),$$

and the volatility $\sigma$ is allowed to depend on $S_t$ and $t$. If the volatility function satisfies suitable regularity conditions, the Black-Scholes formula gives the option’s price at time $t < T$:

$$P_t = E^*(e^{-r(T-t)}(K - S_T)^+)\bigg| \mathcal{F}_t,$$

where $E^*(\mathcal{F}_t)$ stands for the conditional expectation with respect to the risk neutral probability. It can be proved that with $\sigma = \sigma(S_t, T-t)$, then $P_{T-t} = u(S_{T-t}, T-t)$, where $u$ is the solution of

$$\begin{cases}
\partial_t u - \frac{\sigma^2}{2} \partial_{xx} u - rx \partial_x u + ru = 0, & \text{for } x > 0, 0 < t \leq T, \\
u(x, 0) = (K - x)^+. 
\end{cases}$$

(2)

The variational formulation of (2) consists of finding a continuous function $u$ defined on the time interval $[0, T]$ with value in a Hilbert space $V$ (see (4) below) such that

$$\frac{d}{dt}(u, w) + a_t(u, w) = 0 \text{ for a.a. } t \in (0, T), \forall w \in V, \quad u(x, 0) = (K - x)^+,$$

(3)

where

$$V = \left\{ v \in L^2(\mathbb{R}_+) : x \frac{dv}{dx} \in L^2(\mathbb{R}_+) \right\},$$

$$a_t(u, w) = \left( \frac{\sigma^2}{2} \partial_x u, \partial_x w \right) + \left( x \partial_x u, (\sigma^2 + x \sigma \partial_x \sigma - r) w \right) + (ru, w),$$

(4)

$$\langle u, w \rangle = \int_0^\infty u(x)w(x)dx.$$
Problem (3) is obtained by multiplying (2) by $w(x)$, integrate on $\mathbb{R}^+$ and integrate by part the term containing $\partial_x u$. The precise meaning of (3) and the conditions for (3) to have a unique solution are subtle. The choice of the space $V$ is dictated by the following: the integrals in equation (3) must exist, and the bilinear form $a_t$ must be continuous on $V \times V$ uniformly in time and satisfy Gårding’s coercivity inequality, i.e. that there exist constants $c$ and $\lambda$, $c > 0$ and $\lambda \geq 0$ such that

$$a_t(v, v) \geq c\|v\|_V^2 - \lambda\|v\|_{L^2(\mathbb{R}^+)}^2 \quad \forall v \in V.$$ 

**Proposition 1.1** If $\sigma$ is a continuous function such that for all $x, t, \sigma(x, t) \in [\sigma_m, \sigma_M]$, $\sigma_m > 0$, and if $x \mapsto x\sigma$ is Lipschitz continuous with a Lipschitz constant independent of $t$, then (3) has a unique solution.

The simplest finite element methods are discrete versions of (3) obtained by replacing $V$ with a finite dimensional space $V_h$, i.e. finding a continuous function $u_h$ defined on the time interval $[0, T]$ with value in the finite dimensional space $V_h \subset V$ such that

$$\frac{d}{dt}(u_h, w_h) + a_t(u_h, w_h) = 0 \quad \text{for a.a. } t \in (0, T) \quad (5)$$

If $\{w^i\}_1^N$ is a basis of $V_h$ then (5) is equivalent to

$$\frac{d}{dt}\left(\sum_{j=1}^N u_j w^j, w^i\right) + a_t\left(\sum_{j=1}^N u_j w^j, w^i\right) = 0 \quad \forall i = 1, \ldots, N, \quad (6)$$

which is a system of differential equations

$$B \frac{dU}{dt} + A(t)U = 0, \quad \text{where } B_{ij} = (w^j, w^i), \quad A_{ij}(t) = a_t(w^j, w^i). \quad (7)$$

A discrete time stepping scheme has still to be applied to (7), for instance the Euler implicit scheme

$$B \frac{U^{m+1} - U^m}{\delta t_m} + A^{m+1}U^{m+1} = 0, \quad (8)$$

where $A^m = A(t_m)$. The easiest choice is to take $V_h$ as a space of piecewise linear functions, (linear finite elements), i.e.

$$V_h = \{v_h \text{ continuous, linear on each subinterval } [x_{i-1}, x_i], \quad v_h(L) = 0\}$$

where $(x_i)_{i=1, \ldots, N}$ is an increasing sequence in the interval $[0, L] \approx \mathbb{R}^+$, with $x_1 = 0$ and $x_N = L$. Then a good choice for $w^i$ is to be the (hat)
function of $V_h$ which is equal to one at $x_i$ and zero at $x_j$, $j \neq i$ (see [1]). With this basis, called the nodal basis, the integrals $B_{ij}$ and $A_{ij}$ can be computed exactly, up to quadrature errors due to $\sigma$ if it is not piecewise polynomial.

It is easy to show that the matrices $A$ and $B$ are tridiagonal so that a resolution of the linear system (8) at each time step is best done with a Gaussian elimination method or a LU factorization. In the end the computational complexity is the same as for a finite difference method with either an explicit or an implicit time scheme.

Convergence in the $V$-norm can be proved to be of the order of $h + \delta t$ ($h^2 + \delta t$ in the $L^2$-norm); it is possible to improve it by using Crank-Nicolson scheme and higher order polynomials instead of the linear ones.

Most interesting is the following a posteriori estimate:

\textbf{Proposition 1.2}

\[ ||u - u_{h,\delta t}||(t_n) \leq c(u_0) \delta t + c \mu \sigma^2 m_0 \sum_{m=1}^n \delta t_m \prod_{i=1}^{m-1} (1 - 2\lambda \delta t_i) \sum_{\omega \in \mathcal{T}_{mh}} \eta_{m,\omega}^2 \]

where $\mu$ is the continuity constant of $a_t$, $\rho = \max_{2 \leq n \leq N} \frac{\delta t_n}{\delta t_{m-1}}$, $\mathcal{T}_{mh}$ is the partition of $(0, L)$ in small intervals $\omega$ at time $t_m$ and

\[ \eta_m^2 = \delta t_m e^{-2\lambda \delta t_{m-1}} \frac{\sigma_m^2}{2} |u^m_h - u^{m-1}_h|^2 \]

\[ \eta_{m,\omega} = \frac{h_\omega}{x_{\max}(\omega)} \left\| \frac{u^m_h - u^{m-1}_h}{\delta t_m} - r x \frac{\partial u^m_h}{\partial x} + ru^m_h \right\|_{L^2(\omega)}, \]

and $h_\omega$ is the diameter of $\omega$.

Note that everything is known after the computation of $u_h$; the mesh can then be adapted by using the error indicators $\eta_m$ and $\eta_{m,\omega}$. Figure 1 shows the computed a posteriori error and the actual error for a vanilla put with constant volatility and comparison with the Black-Scholes analytic formula.

\section{Stochastic Volatility Models}

We now discuss a more involved stochastic volatility model where $dS_t = S_t(\mu dt + \sigma_t dW_t)$, with $\sigma_t = f(Y_t)$ and $Y_t$ is a mean-reverting Ornstein-Uhlenbeck process

\[ dY_t = \alpha (m - Y_t) dt + \beta d\hat{Z}_t \]
Figure 1: The first graph displays the error between the computed solution on a uniform mesh and one computed with Black-Scholes formula. The second graph show $\rho_{rt}$; the third graphs shows $\eta_{n,\omega}$ as a function of $x$ and $t$. The parameters are: $K = 100, T = 0.5, r = 0, \sigma = 0.3$, 50 time steps and 100 mesh points.

and $\tilde{Z}_t = \rho W_t + \sqrt{1 - \rho^2} Z_t$, $W_t$ and $Z_t$ are two uncorrelated Brownian motions. In the Stein-Stein model, see [4], $f(Y_t) = |Y_t|$. If $P(S, y, T - t)$ is the pricing function of a put option, it is convenient to introduce

$$u(S, y, t) = P(S, y, T - t)e^{-(1-\eta)\frac{(y-m)^2}{2x^2}},$$

where $\eta$ is any parameter between 0 and 1 and $\nu^2 = \beta^2/(2\alpha)$.

A clever financial argument (see [2]) shows that when $\rho = 0$ for simplicity, the new unknown $u$ satisfies, for all $x > 0$, $y \in \mathbb{R}, t > 0$

$$
\begin{cases}
\frac{\partial u}{\partial t} - \frac{1}{2} y^2 \frac{\partial^2 u}{\partial y^2} - \frac{1}{2} \beta^2 \frac{\partial^2 u}{\partial y^2} - rx \frac{\partial u}{\partial x} + e \frac{\partial u}{\partial y} + fu = 0, \\
u(x, y, 0) = (K - x)^+, \quad \text{for} \quad x > 0, y \in \mathbb{R},
\end{cases}
$$

where

$$e = -(1 - 2\eta)\alpha(y - m) + \beta \gamma$$

$$f = r + 2\alpha^2 \eta(1 - \eta)(y - m)^2 + 2(1 - \eta)\frac{\alpha^2}{2}(y - m)\gamma - \alpha(1 - \eta)$$

and $\gamma$ is a risk premium factor, possibly a function of $x, y, t$, see [7] for the details.

This parabolic equation is degenerate near the axis $y = 0$ because the coefficient in front of $x^2 \frac{\partial^2 P}{\partial x^2}$ vanishes on $y = 0$. Following [7], see also [1, 8], let us consider the weighted Sobolev space $V$ on $Q = \mathbb{R}^+ \times \mathbb{R}$:

$$V = \left\{ v : v\sqrt{1 + y^2} \frac{\partial v}{\partial y}, x|y| \frac{\partial v}{\partial x} \in L^2(Q) \right\},$$

which is a Hilbert space such that all the properties needed by the variational arguments sketched above hold.
Theorem 2.1 If the data are smooth, (see [7, 8]) and if
\[ r(t) \geq r_0 > 0, \quad \alpha^2 > 2\beta^2, \quad 2\alpha^2\eta(1 - \eta) > \beta^2, \]
then (10) has a unique weak solution.

2.1 Finite Element Solution

As usual we localize the problem in \( \Omega := (0, L_x) \times (-L_y, L_y) \). No boundary condition is needed on the axis \( x = 0 \) since the PDE is degenerate there. Notice also that \( u \) is expected to tend to zero as \( x \) tends to infinity. For large \( y \) no arguments seems to give a boundary condition. However, if \( L_y \) is chosen such that \( \alpha(L_y \pm m) \gg \beta^2 \), then for \( y \sim \pm L_y \), the coefficient of the advection term in the \( y \) direction, \( \alpha(y - m) \), is much larger in absolute value than the coefficient of the diffusion in the \( y \) direction, \( \beta^2 \). Furthermore, near \( y = \pm L_y \), the vertical velocity \( \alpha(y - m) \) is directed outward \( \Omega \). Therefore, the error caused by an artificial condition on \( y = \pm L_y \) will be damped away from the boundaries, and localized in boundary layers whose width is of the order of \( \beta^2 \). We may thus apply a Neumann condition.

Then one must write the problem in variational form: find \( u \) in \( V_0 \) the subset of \( V \) of functions which are zero at \( x = L_x \) and

\[
\begin{aligned}
\frac{d}{dt}(u, w) + a(u, w) &= 0, \quad \forall w \in V_0, \\
u(x, y, 0) &= \text{given.}
\end{aligned}
\] (13)

with

\[
a(u, w) = \int_{\Omega} \left( \frac{y^2}{2} \partial_x u \partial_x w + \frac{\beta^2}{2} \partial_y u \partial_y w + (y^2 - r)x \partial_x u + (e + \beta \partial_y \beta) \partial_y u + fu \right) w \] (14)

Now \( \Omega \) is triangulated into a set of non-intersecting triangles \( \{ \omega_k \}_{1}^{K} \) in such a way that no vertices lie in the middle of an edge of another triangle. \( V_0 \) is approximated by \( V_{0h} \) the space of continuous piecewise linear functions on the triangulation which vanish at \( x = L_x \).

The functions of \( V_{0h} \) are completely determined by their values at the vertices of the triangles, so if we denote by \( w^i \) the function in \( V_{0h} \) which takes the value zero at all vertices except the \( i^{th} \) one where it is one, then \( \{ w^i \}_{1}^{N} \) is a basis of \( V_{0h} \) if \( N \) is the number of vertices not on \( x = L_x \) (it is convenient for the presentation to number the nodes on the boundary last).
Table 1: Comparison of CPU time for the LU algorithm and superLU for a product put option on a uniform mesh and 200 step time (computed by N. Lantos).

<table>
<thead>
<tr>
<th>Mesh size</th>
<th>Gauss-LU [s]</th>
<th>Relative error</th>
<th>superLU [s]</th>
<th>Relative error</th>
</tr>
</thead>
<tbody>
<tr>
<td>101 × 101</td>
<td>10.094</td>
<td>3.532 %</td>
<td>2.39</td>
<td>3.076 %</td>
</tr>
<tr>
<td>126 × 126</td>
<td>14.547</td>
<td>1.338 %</td>
<td>4.016</td>
<td>1.797 %</td>
</tr>
<tr>
<td>151 × 151</td>
<td>22.313</td>
<td>0.751 %</td>
<td>6.203</td>
<td>0.489 %</td>
</tr>
<tr>
<td>176 × 176</td>
<td>31.985</td>
<td>1.131 %</td>
<td>8.735</td>
<td>0.790 %</td>
</tr>
<tr>
<td>201 × 201</td>
<td>43.938</td>
<td>0.432 %</td>
<td>12.109</td>
<td>0.670 %</td>
</tr>
</tbody>
</table>

As in the one dimensional case

\[ u_m^m(x, y) = \sum_{i=1}^{N} u_i^m w_i(x, y) \]

is an approximation of \( u(t_m) \) if the \( u_i^{m+1} \) are solutions of

\[ B \frac{U^{m+1} - U^m}{\delta t_m} + AU^{m+1} = 0 \]  \( \text{(15)} \)

with \( B_{ij} = (w^j, w^i) \) and \( A_{ij} = a(w^j, w^i) \).

**2.2 Numerical Implementation**

Integrals of polynomial expressions of \( x, y \) can be evaluated exactly on triangles (formula (4.19) in [1]). The difficulty is in the strategy to solve the linear systems (15). Multigrid seems to give the best results in term of speed ([9]) but the computer program may not be so easy to solve for an arbitrary triangulation. Since the number of vertices are not likely to exceed a few thousands, fast multifrontal libraries such as SuperLU [10] can be used with a minimum programming effort. Table 2.2 shows the gain in time over more classical algorithms.

Alternatively one could also use a Preconditioned Conjugate Gradient -like iterative method (see eqf12/005 in this book). Another way is to use FreeFem++[11] which is a general two dimensional PDE solver particularly well suited to these option pricing models where the coefficients are involved functions and one may want to try to change them often. Results for (15) obtained with the following FreeFem++ code are on figure 2:

```c
real T=1, K=100, r=0.05, alpha=1, beta=1, m=0.2, gamma=0, eta=0.5;
int Lx=800, Ly=3, Nmax=50;
real scale=100, dt = T/Nmax;
```
mesh th = square(50,50,[x*Lx,(2*y-1)*Ly*scale]);
fespace Vh(th,P1);

func u0 = max(K-x,0.)*exp(-(1-eta)*(y/scale-m)^2 * alpha/(beta^2));
func e = beta*gamma -(1-2*eta)*alpha*(y/scale-m);
func f = r + 2*(alpha/beta)^2 * eta*(1-eta)*(y/scale-m)^2
     + 2*(1-eta)*(alpha/beta)*(y/scale-m)*gamma - alpha*(1-eta);

Vh uold=u0,u,v;

problem stein(u,v,init=n) = int2d(th)( u*v*(f+1/dt)
   + dx(u)*dx(v) *(x*y/scale)^2/2 + dy(u)*dy(v) *(beta*scale)^2/2
   + dx(u)*v *x*((y/scale)^2-r) + dy(u)*v *e*scale
   ) - int2d(th)(uold*v/dt) + on(2,u=0) ;

for (int n=0; n<Nmax ; n++){ stein; uold=u; }
// call to keyword adaptmesh not shown....

Figure 2: Solution of (10) for a put with maturity T=1, strike K=100 with
r = 0.02, β = α = 1 and m = 0.2. We have used the automatic mesh
adaptivity of FreeFem++; the contours and the triangulation are shown.

A similar program has been developed in 3D by S. Delpino: freefem3d.
However mesh refinement is not yet implemented. A basket option has been
computed for illustration (figure 3). The script that drives the program is
self explanatory and gives the values used.

double N = 25; double L=200.0; double T=0.5;

8
double dt = T / 15; double K = 100; double r = 0.02;
double s1 = 0.3; double s2 = 0.2; double s3 = 0.25;
double q12 = -0.2 * s1 * s2; double q13 = -0.1 * s1 * s3;
double q23 = -0.2 * s2 * s3; double s11 = s1 * s1 / 2;
double s22 = s2 * s2 / 2; double s33 = s3 * s3 / 2;

vector n = (N, N, N);
vector a = (0, 0, 0);
vector b = (L, L, L);
mesh M = structured(n, a, b);
femfunction uold(M) = max(K - x - y - z, 0);
femfunction u(M);

double t = 0; do{
    solve(u) in M cg(maxiter=900, epsilon=1E-10)
    {
pde(u) (1./dt+r)*u - dx(s11*x*x*dx(u)) - dy(s22*y*y*dy(u)) - dz(s33*z*z*dz(u))
        - dx(q12*x*y*dy(u)) - q13*x*z*dz(u)) - r*x*dx(u) - r*y*dy(u) - r*z*dz(u) = uold/dt;
        dnu(u) = 0 on M;
    }
    t = t + dt;
} while(t < T);
save(medit, "u", u, M); save(medit, "u", M);

Figure 3: A put on a basket of 3 assets computed with freefem3d[12] with maturity T = 0.5, strike K = 100 etc (see the listing). The visualization is done with medit also public domain [13]
3 Asian Calls

For the financial modeling of Asian options, we refer to [14] and the references therein.

We consider an Asian put with fixed strike whose payoff is $P_0(S, y) = (y - K)^+$, calling $y$ the average value of the asset in time, $y = \frac{1}{T} \int_0^T S(\tau) d\tau$.

The price of the option is found by solving for all $\{x, y, t\} \in \mathbb{R}^+ \times \mathbb{R}^+ \times (0, T]$

$$
\partial_t u - \partial_x \left( \frac{\sigma^2 x^2}{2} \partial_x u \right) + (\sigma^2 - r)x \partial_x u + \frac{y - x}{T - t} \partial_y u + ru = 0,
$$

$$
u(x, y, 0) = (y - K)^+, \hspace{1cm} \partial_x u|_{+\infty,y,t} \approx \begin{cases} 
\frac{1}{T-t}e^{-rt} & \text{if } y < \frac{KT}{T-t} \\
 \frac{1}{T-t}r e^{-rt} & \text{otherwise}.
\end{cases}
$$

(16)

The difficulties of this problem is that the second order part of the differential operator is incomplete and the convective velocity $v = ((\sigma^2 - r)x, \frac{y-x}{T-t})^T$ tends to infinity as $t \to T$; so great care must be taken for the upwinding of these first order terms. One of the best upwinding is the characteristic finite element method whereby

$$
\partial_t u + v \cdot \nabla u|_{x,y,t,m+1} \approx \frac{1}{\delta t}(u^{m+1}(q) - u^m(Q)),
$$

with $(Q = q - \delta t v(q - v(q) \delta t / 2))$ where $q = (x, y)^T$.

(17)

The discontinuous $P^2$ reconstruction with characteristic convection of the gradient and of the value at the center of gravity is applied. The domain is localized to $(0, L) \times (0, L)$, with $L = 250$, i.e. greater than twice the contract price $K = 100$, in this example; the interest rate is $r = 3\%$ and the volatility $\sigma = 0.3$. The square domain is triangulated by a uniform 50 $\times$ 50 mesh and we have taken 50 time steps over the 4 years period of this example ($T=4$). The results are shown on figure 4 at time $t = 0.96, 1.96, 3.96$.

References


Figure 4: Solution of (16) for an Asian option of maturity \( T = 4 \) and strike \( K = 100 \) when the interest rate is 0.03 and the volatility 0.3. The three figures show the contours at times 0.96, 1.96 and 3.96.


    www.ann.jussieu.fr/~frey/logiciels/medit.html