MATHEMATISCHES INSTITUT
DER UNIVERSITÄT ZU KÖLN
Prof. Dr. Rüdiger Seydel
Dipl.-Math. Christian Jonen

Winter 2009/10
November 12

Computational Finance 2 - 5th Assignment

Deadline: November 19

Exercise 10 (Front Fixing Finite Element Approach II) (3+6 Points)

Exercise 9 leads to the following nonlinear problem:

$$u_\tau - \gamma u_{xx} + (\nu - b'(\tau))u_x + \mu u = 0,\ 0 < x < L,\ 0 < \tau \le T,$$

$$u(0,\tau) = 0,\ 0 \le \tau \le T,$$
$$u(L,\tau) = g(b(\tau),\tau),\ 0 \le \tau \le T,$$
$$u_x(L,\tau) = g_y(b(\tau),\tau),\ 0 \le \tau \le T,$$
$$u(x,0) = u_0(x),\ 0 < x < L,$$

where $u_0 = g(a(0)+x,0)$. The idea is to integrate the Neumann boundary condition into the variational problem and use the Dirichlet boundary condition to determine the free boundary $b(\tau)$.

a) The variational form for the problem above is as follows:
Find $u \in L^2(0,T;H^1_E(\Omega))$ and $b \in C([0,T]) \cap C^1((0,T])$ such that $u_\tau \in L^2(0,T;H^{-1}(\Omega))$, $u(0) = u_0$ and

$$(u_\tau, v) + \mathcal{B}(b'(\tau),u,v) - f(b(\tau),\tau)v(L) = 0 \quad \forall v \in H^1_E(\Omega),\ 0 < \tau \le T,$$

$$u(L,\tau) = g(b(\tau),\tau),\ 0 < \tau \le T,$$

where $f(y,\tau) = \gamma g_y(y,\tau)$; $L^2(\Omega)$ denotes the space of square integrable functions on $\Omega = (0,L)$, and $(\cdot,\cdot)$ denotes the inner product of $L^2(\Omega)$. Furthermore, $H^1_E(\Omega)$ is the closure of $\{v \in C^\infty(\overline{\Omega}) : v(0) = 0\}$ in the usual Sobolev space $H^1(\Omega)$, and $H^{-1}(\Omega)$ is the dual space of $H^1_E(\Omega)$. Define the bilinear form $\mathcal{B}(b'(\tau),u,v)$.

b) Approximate the variational problem by finite elements. Proceed as follows:

- Find an appropriate grid.
- Use the Galerkin approach with hat functions. In so doing, let V_h be the piecewise linear element subspace of $H^1_E(\Omega)$ with respect to your grid. Work with the ansatz $w := \sum_{i=0}^{N} c_i\phi_i$, where N is the number of intervals with respect to the x-grid. Find an appropriate time discretization scheme.
- Convert your finite element approximation to a matrix form.

Exercise 11 (Implementation of a Finite Element Approach) (3+6 Points)

A comfortable way of solving PDEs by finite element approaches is to use ready-made classes or programs. FreeFEM++ is able to deal with such problems and is based on C++. Download freeFEM++ at

$$www.freefem++.org$$

In order to test the functionality of freeFEM++, implement the following codes:

- ```
  bool debug = true;
  mesh Th = square(10,10);
  ```

- ```
  bool debug = true;
  border C(t=0,2*pi)x=cos(t);y=sin(t);
  mesh Th=buildmesh(C(10));
  fespace Vh(Th,P1);
  plot(Th);
  Vh u,v;
  func f=-1;
  problem Poisson(u,v)=int2d(Th)(dx(u)*dx(v)+dy(u)*dy(v))+int2d(Th)(-f*v)
     +on(C,u=0);
  Poisson;
  ```

Save the codes as .edp files. The first code generates a grid. Describe the structure of the grid by adding the command

plot(Th);

The second code solves the Poisson equation with Dirichlet boundary condition:

$$\begin{aligned} -\Delta u &= f \text{ in } \Omega \subset \mathbb{R}^2,\ \Delta = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}, \\ u &= 0 \text{ on } \partial\Omega. \end{aligned}$$

Plot the structure of the grid and the solution by adding the command

$$plot(u);$$

Comment both codes.