A short introduction to FreeFem++

M. Ersoy

Basque Center for Applied Mathematics

12 November 2010

Outline of the talk

1. Introduction

2. HOW TO
   - solve steady pdes : e.g. Laplace equation
   - solve unsteady pdes : e.g. Heat equation

Outline

1. Introduction

2. HOW TO
   - solve steady pdes : e.g. Laplace equation
   - solve unsteady pdes : e.g. Heat equation

A software for solving PDEs

FreeFem++ for 2D-3D[1] PDEs

- Finite element method

1. in progress

A software for solving PDEs

FreeFem++ for 2D-3D [1] PDEs

- Finite element method
- Free software : available at `http://www.freefem.org/ff++/` [2] , it runs on
  - Linux
  - Windows
  - Mac

1. in progress
2. Useful documentation available at `http://www.freefem.org/ff++/ftp/freefem++doc.pdf`

A SOFTWARE FOR SOLVING PDEs

FreeFem++ for 2D-3D [1] PDEs

- Finite element method
- Free software : available at `http://www.freefem.org/ff++/` [2] , it runs on
  - Linux
  - Windows
  - Mac
- ++ extension of FreeFem and FreeFem+ (see Historic `http://www.freefem.org/ff++/ftp/HISTORY`)

1. in progress
2. Useful documentation available at `http://www.freefem.org/ff++/ftp/freefem++doc.pdf`

A software for solving PDEs

FreeFem++ for 2D-3D [1] PDEs

- Finite element method
- Free software : available at http://www.freefem.org/ff++/ [2] , it runs on
  - Linux
  - Windows
  - Mac
- ++ extension of FreeFem and FreeFem+ (see Historic http://www.freefem.org/ff++/ftp/HISTORY)
- FreeFem++ team : Olivier Pironneau, Frédéric Hecht, Antoine Le Hyaric, Jacques Morice

1. in progress
2. Useful documentation available at http://www.freefem.org/ff++/ftp/freefem++doc.pdf

Outline

1. Introduction

2. HOW TO
   - solve steady pdes : e.g. Laplace equation
   - solve unsteady pdes : e.g. Heat equation

OUTLINE

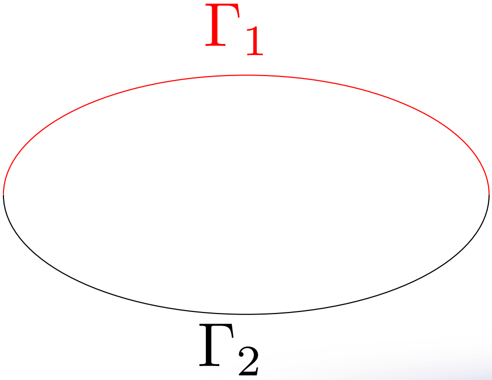
1. INTRODUCTION

2. HOW TO
   - solve steady pdes : e.g. Laplace equation
   - solve unsteady pdes : e.g. Heat equation

The problem

Given $f \in L^2(\Omega)$, find $\varphi \in H_0^1(\Omega)$ such that :

$$\begin{cases} -\Delta\varphi = f & \text{in } \Omega \\ \varphi(x_1, x_2) = 0 & \text{on } \Gamma_1 \\ \partial_n \varphi := \nabla\varphi \cdot n = 0 & \text{on } \Gamma_2 \end{cases}$$



Variational formulation (VF)

Let w be a test function, the VF of the Laplace equation is :

$$\int_\Omega \nabla\varphi \cdot \nabla w \, dx = \int_\Omega f w \, dx$$

or equivalently

$$A(\varphi, w) = l(w)$$

where the bilinear form is

$$A(\varphi, w) = \int_\Omega \nabla\varphi \cdot \nabla w \, dx$$

and the linear one :

$$l(w) = \int_\Omega f w \, dx$$

Variational problem

Problem is now to find $v \in H_0^1(\Omega)$ such that, for every $w \in H_0^1(\Omega)$:

$$A(\varphi, w) = l(w)$$

where the bilinear form is

$$A(\varphi, w) = \int_\Omega \nabla\varphi \cdot \nabla w \, dx$$

and the linear one :

$$l(w) = \int_\Omega f w \, dx$$

We set $V = H^1(\Omega)$ in the next

FEM : Galerkin method

Let $V_h \subset V$ with $\dim V_h = n_h$,

$$V_h = \left\{ u(x,y);\ u(x,y) = \sum_{k=1}^{n_h} u_k \phi_k(x,y),\ u_k \in \mathbb{R} \right\}$$

where $\phi_k \in P_s$ is a polynom of degree s.
Now, the problem is to find $\varphi_h \in V_h$ such that :

$$\forall w \in V_h, A(\varphi_h, w) = l(w)$$

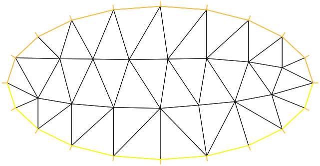
So, we have to solve :

$$\forall i \in \{1, \ldots, n_h\}, A(\phi_j, \phi_i) = l(\phi_i)$$

Spaces V_h

Spaces $V_h = V_h(\Omega_h, P)$ will depend on the mesh :

$$\Omega_h = \bigcup_{k=1}^{n} T_k$$



and the approximation P

- P_0 piecewise constant approximation
- P_1 C^0 piecewise linear approximation
- $\vdots$

With FreeFem++

To numerically solve the Laplace equation, we have to :

1. mesh the domain (define the border + mesh)
2. write the VF
3. show the result

it's easy !

Step 1 : mesh of the domain

if $\partial\Omega = \Gamma_1 + \Gamma_2 + \ldots$ then

- FreeFem++ define border commands :
  - border Gamma1(t = t0,tf){x = gam11(t), y = gam12(t)} ;
  - border Gamma2(t = t0,tf){x = gam21(t), y = gam22(t)} ;
  - ...

  where the set {x = gam11(t), y = gam12(t)} is a parametrisation of the border Gamma1 with $t \in [t0, tf]$

Step 1 : mesh of the domain

if $\partial\Omega = \Gamma_1 + \Gamma_2 + \dots$ then

- FreeFem++ define border commands :
  - border Gamma1(t = t0,tf){x = gam11(t), y = gam12(t)} ;
  - border Gamma2(t = t0,tf){x = gam21(t), y = gam22(t)} ;
  - ...

  where the set {x = gam11(t), y = gam12(t)} is a parametrisation of the border Gamma1 with $t \in [t0, tf]$
- FreeFem++ mesh commands :

  mesh MeshName = buildmesh(Gamma1(m1)+Gamma1(m2)+. . .) ;

  where mi are positive (or negative) numbers to indicate the number of point should on Γ_j. example

To write the VF, we need to

- define finite element space , we will use :

  fespace NameFEspace(MeshName,P) ;

  where P = P0 or P1 or P2, ... Example :

  fespace Vh(Omegah,P2) ;

To write the VF, we need to

- define finite element space , we will use :

  fespace NameFEspace(MeshName,P) ;

  where P = P0 or P1 or P2, ... Example :

  fespace Vh(Omegah,P2) ;

- use interpolated function, for instance,

  Vh f = x+y ;

  N.B. x and y are reserved key words

Step 2 : write the VF

VF

VF is simply what we write on paper, for instance, for the Laplace equation, we should have :

```
Vh phi, w, f ;

problem Laplace(phi,w)
int2d(Omegah)(dx(phi)*dx(w) + dy(phi)*dy(w))
- int2d(Omegah)(f*w)
+ boundary conditions
;
```

where w, phi belong to FE space Vh.

Step 2 : write the VF

VF

VF is simply what we write on paper, for instance, for the Laplace equation, we should have :

```
Vh phi, w, f ;

problem Laplace(phi,w)
int2d(Omegah)(dx(phi)*dx(w) + dy(phi)*dy(w))
- int2d(Omegah)(f*w)
+ boundary conditions
;
```

where w, phi belong to FE space Vh.
Next, to solve it, just write :

```
Laplace;
```

Step 2 : write the VF

VF

VF is simply what we write on paper, for instance, for the Laplace equation, we should have :

```
Vh phi, w, f ;

problem Laplace(phi,w)
int2d(Omegah)(dx(phi)*dx(w) + dy(phi)*dy(w))
- int2d(Omegah)(f*w)
+ boundary conditions
;
```

where w, phi belong to FE space Vh.

or equivalently write directly :

```
solve Laplace(phi,w)
int2d(Omegah)(dx(phi)*dx(w) + dy(phi)*dy(w))
- int2d(Omegah)(f*w)
+ boundary conditions
;
```

Step 2 : write the VF

Boundary conditions [3]

- Dirichlet condition $u = g$: +on(BorderName, u=g)
- Neumann condition $\partial_n u = g$: -int1d(Th)(g*w)
- ...

3. see the manual p. 142

Step 3 : show the result

to plot

```
plot(phi);
```

or

```
plot([dx(phi),dy(phi)]);
```

Laplace equation

1. INTRODUCTION

2. HOW TO
   - solve steady pdes : e.g. Laplace equation
   - solve unsteady pdes : e.g. Heat equation

The problem

Given $f \in L^2(\Omega)$, find $\varphi \in H^1(\Omega)$ such that :

$$\begin{cases} \partial_t \varphi - \Delta\varphi = f & \text{in } \Omega \\ \varphi(x_1, x_2) = z(x_1, x_2) & \text{on } \Gamma_1 \\ \partial_n \varphi := \nabla\varphi \cdot n = 0 & \text{on } \Gamma_2 \end{cases}$$

may be rewritten, after an implicit Euler finite difference approximation in time as follows :

$$\frac{\varphi^{n+1} - \varphi^n}{\delta t} - \Delta\varphi^{n+1} = f^n$$

Variational formulation

The Variational formulation for the semi discrete equation is : let w be a test function, we write :

$$\int_\Omega \frac{\varphi^{n+1} - \varphi^n}{\delta t} w + \nabla \varphi^{n+1} \cdot \nabla w \, dx = \int_\Omega f^n w \, dx$$

The "FreeFem formulation"

Following the steady state case, the problem is then :

```
solve Laplace(phi,w)
int2d(Omegah)(phi*w/dt +dx(phi)*dx(w) + dy(phi)*dy(w))
- int2d(Omegah)(phiold*w/dt f*w)
+ boundary conditions
;
```

where we have to solve iteratively this discrete equation. example


Enjoy
yourself