FreeFem++, New Finite Elements, New Graphics, ...

F. Hecht

Laboratoire Jacques-Louis Lions
Université Pierre et Marie Curie
Paris, France

with O. Pironneau, J. Morice

http://www.freefem.org mailto:hecht@ann.jussieu.fr

With the support of ANR (French gov.) ANR-07-CIS7-002-01

http://www.freefem.org/ff2a3/ http://www-anr-ci.cea.fr/

PLAN

- Introduction Freefem++
- 3d Finite element
- 3d intégration
- 3d plot
- Matrix
- Array operator

http://www.freefem.org/

3d Finite element

Currently `FreeFem++` implements the following elements in 3d :

`P03d` piecewise constant,

`P13d` continuous piecewise linear,

`P23d` continuous piecewise quadratic,

`RT03d` Raviart-Thomas piecewise constant,

`Edge03d` The Nedelec Edge element

`P1b3d` piecewise linear continuous plus bubble,

...

3d Finite element

P0,P03d piecewise constante discontinuous finite element (2d, 3d), the degres of freedom are the barycenter element value.

$$P0_h = \left\{ v \in L^2(\Omega) \,\middle|\, \text{for all } K \in \mathcal{T}_h \text{ there is } \alpha_K \in \mathbb{R} : \quad v_{|K} = \alpha_K \right\} \tag{1}$$

The degree of freedom is the value at the barycenter

P1,P13d piecewise linear continuous finite element (2d, 3d), the degres of freedom are the vertices values.

$$P1_h = \left\{ v \in H^1(\Omega) \,\middle|\, \forall K \in \mathcal{T}_h \quad v_{|K} \in P_1 \right\} \tag{2}$$

The degree of freedom are the value at the vertices

3d Finite element

P1b,P1b3d piecewise linear continuous finite element plus bubble (2d, 3d)

$$P1b_h = \left\{ v \in H^1(\Omega) \,\middle|\, \forall K \in \mathcal{T}_h \quad v_{|K} \in P_1 \oplus \mathsf{Span}\{\lambda_0^K \lambda_1^K \lambda_2^K \lambda_3^K\} \right\} \qquad (3)$$

where $\lambda_i^K, i = 0, .., d$ are the $d + 1$ barycentric coordinate functions of the element K (triangle or tetrahedron).

The degree of freedom are the value at the vertices and at the barycenter

P2,P23d piecewise P_2 continuous finite element (2d, 3d),

$$P2_h = \left\{ v \in H^1(\Omega) \,\middle|\, \forall K \in \mathcal{T}_h \quad v_{|K} \in P_2 \right\} \qquad (4)$$

where P_2 is the set of polynomials of $\mathbb{R}^2$ of degrees ≤ 2.

The degree of freedom are the value at the vertices and at the middle of edges

3d Finite element

RT03d Raviart-Thomas finite element of degree 0. The 3d case :

$$RT0_h = \left\{ \mathbf{v} \in H(\mathrm{div}) \middle| \forall K \in \mathcal{T}_h \quad \mathbf{v}_{|K}(x,y,z) = \left| \begin{matrix} \alpha_K^1 \\ \alpha_K^2 \\ \alpha_K^3 \end{matrix} \right. + \beta_K \left| \begin{matrix} x \\ y \\ z \end{matrix} \right. \right\} \qquad (5)$$

where by writing div $\mathbf{w} = \sum_{i=1}^{d} \partial w_i / \partial x_i$ with $\mathbf{w} = (w_i)_{i=1}^{d}$,

$$H(\mathrm{div}) = \left\{ \mathbf{w} \in L^2(\Omega)^d \middle| \mathrm{div}\ \mathbf{w} \in L^2(\Omega) \right\}$$

and where $\alpha_K^1, \alpha_K^2, \alpha_K^3, \beta_K$ are real numbers. The degree of freedom is the integral of the normal componante on each triangulare face, computer the by formula with the 3 points middle of edges. The sign of the normal is choose the the face numbering.

3d Finite element

Edge03d Nedelec finite element or Edge Element of degree 0.

$$N0_h = \left\{ \mathbf{v} \in H(\text{Curl}) \middle| \forall K \in \mathcal{T}_h \quad \mathbf{v}_{|K}(x,y,z) = \left| \begin{array}{l} \alpha^1_K \\ \alpha^2_K \\ \alpha^3_K \end{array} + \right| \begin{array}{l} \beta^1_K \\ \beta^2_K \\ \beta^3_K \end{array} \times \left| \begin{array}{l} x \\ y \\ z \end{array} \right. \right\} \tag{6}$$

where by writing $\text{curl}\mathbf{w} = \left| \begin{array}{l} \partial w_2/\partial x_3 - \partial w_3/\partial x_2 \\ \partial w_3/\partial x_1 - \partial w_1/\partial x_3 \\ \partial w_1/\partial x_2 - \partial w_2/\partial x_1 \end{array} \right.$ with $\mathbf{w} = (w_i)_{i=1}^d$,

$$H(\text{curl}) = \left\{ \mathbf{w} \in L^2(\Omega)^d \middle| \text{curl } \mathbf{w} \in L^2(\Omega)^d \right\}$$

and $\alpha^1_K, \alpha^2_K, \alpha^3_K, \beta^1_K, \beta^2_K, \beta^3_K$ are real numbers.

The degree of freedom are the integral of the tangent componante on each edge, computer the by Gauus formula with the 2 points. The sign of the edge is choose the the face numbering.

Array of Mesh, Finite element

```
mesh[int] aTh(3);
aTh[0]=square(3,3);
aTh[1]=square(4,4,[x+1,y]);
plot(aTh,wait=1);                                           //    skip unset meshes
mesh Th=aTh[0];
fespace Vh(Th,P1);
Vh[int] au(3);                         //    array of 2 finite element functions
Th=aTh[0];          au[0] =x;
Th=aTh[1];          au[1] =y;
plot(au,wait=1);                                      //    skip unset FE functions
```

Bug in array of vectorial FE function

3d quadrature

For a three dimensional domain $\Omega_h = \sum_{k=1}^{n_t} T_k$, $\mathcal{T}_h = \{T_k\}$, we can calculate the integral over Ω_h by

$$\begin{aligned}\int_{\Omega_h} f(x,y) &= \texttt{int3d(Th)(f)}\\ &= \texttt{int3d(Th,qfV=*)(f)}\\ &= \texttt{int3d(Th,qforder=*)(f)}\end{aligned}$$

where $*$ stands for the name of quadrature formula or the order of the Gauss formula.

Quadature formula on a tetraedron					
N	`qfV=`	`qforder=`	point in $T_k \in \mathbb{R}^3$	ω_ℓ	exct. P_k
1	qfV1	2	$\left(\frac{1}{4},\frac{1}{4},\frac{1}{4}\right)$	$\|T_k\|$	1
4	qfV2	3	$G4(0.58\ldots,0.13\ldots,0.13\ldots)$	$\|T_k\|/4$	2
14	**qfV5**	6	$G4(0.72\ldots,0.092\ldots,0.092\ldots)$ $G4(0.067\ldots,0.31\ldots,0.31\ldots)$ $G6(0.45\ldots,0.045\ldots,0.45\ldots)$	$0.073\ldots\|T_k\|$ $0.11\ldots\|T_k\|$ $0.042\ldots\|T_k\|$	5
4	qfV1lump		$G4(1,0,0)$	$\|T_k\|/4$	1

Where $G4(a, b, b)$ such that $a + 3b = 1$ is the set of the four point in barycentric coordinate

$$\{(a, b, b, b), (b, a, b, b), (b, b, a, b), (b, b, b, a)\}$$

and where $G6(a, b, b)$ such that $2a + 2b = 1$ is the set of the six points in barycentric coordinate

$$\{(a, a, b, b), (a, b, a, b), (a, b, b, a), (b, b, a, a), (b, a, b, a), (b, a, a, b)\}.$$

Remark, all this tetraedral quadrature formula come from `http://www.cs.kuleuven.be/~nines/research/ecf/mtables.html`

3d plot / via ffglut

The parameters of the plot command can be , meshes, real FE functions , array of meshes, array of FE functions, arrays of 2 real FE functions, arrays of two arrays of double, to plot respectively a mesh, a function, a vector field, or a curve defined by the two arrays of double.

The parameters are

wait= boolean expression to wait or not (by default no wait). If true we wait for a keyboard up event or mouse event, they respond to an event by the following characters

`enter)` wait next plot

`p)` previous plot (10 plots saved)

`ESC)` exit from ffglut

`?)` show this help window

`+) -)` zoom in/out around the cursor 3/2 times

=) reset vue

r) refresh plot

up, down, left, right) special keys to tanslate

3) switch 3d/2d plot (in test) keys :

- z) Z) (focal zoom unzoom)
- H) h) switch increase or decrease the Z scale of the plot

mouse motion)

- - left button) rotate
- - right button) zoom (ctrl+button on mac)
- - right button +alt) tanslate (alt+ctrl+button on mac)

a) A) increase or decrease the arrow size

B) switch between show border meshes or not

i) I) update or not : the min/max bound of the functions to the window

n) N) decrease or increase the number of iso value array

- `b)` switch between black and white or color plotting
- `g)` switch between grey or color plotting
- `f)` switch between filling iso or iso line
- `l)` switch between lighting or not
- `v)` switch between show or not the numerical value of colors
- `m)` switch between show or not meshes
- `w)` window dump in file ffglutXXXX.ppm
- `any other key` do nothing : nothing

`ps=` string expression to save the plot on postscript file

`coef=` the vector arrow coef between arrow unit and domain unit.

`fill=` to fill between isovalues.

`cmm=` string expression to write in the graphic window

`value=` to plot the value of isoline and the value of vector arrow.

`aspectratio=` boolean to be sure that the aspect ratio of plot is preserved or not.

`bb=` array of 2 array (like `[[0.1,0.2],[0.5,0.6]]`), to set the bounding box and specify a partial view where the box defined by the two corner points [0.1,0.2] and [0.5,0.6].

`nbiso=` (int) sets the number of isovalues (20 by default)

`nbarrow=` (int) sets the number of colors of arrow values (20 by default)

`viso=` sets the array value of isovalues (an array real[int] of increasing values)

`varrow=` sets the array value of color arrows (an array real[int])

`bw=` (bool) sets or not the plot in black and white color.

`grey=` (bool) sets or not the plot in grey color.

`hsv=` (array of float) to defined color of 3*n value in HSV color model declare with for example

```
real[int] colors = [h1,s1,v1,... ,  hn,vn,vn];
```

where `hi,si,vi` is the ith color to defined the color table.

`boundary=` (bool) to plot or not the boundary of the domain (true by default).

`dim=` (int) sets dim of the plot 2d or 3d (2 by default)

ffglut tools

`ffglut` is new program freefem++ graphic server base on GLUT/OpenGL libary

The graphic architecture is client/server, freefem is the graphics cleint

if the freefem++ program do not containt '-' character then the graphic server is launch trough a pipe. All the graphical data are send with is pipe.

so you can save all the graphics in batch mode

```
FreeFem++  -fglut /tmp/ff-graphics  my.edp
ffglut /tmp/ff-graphics
```

So new FreeFem++ is now `FreeFem++-nw`

Under Window I a had a small `fflaunch.exe`

Remaks on formal Array operator

I have add lot of formal operator on array or matrices of type [...].

*,+,- product,sum, difference

.*,./ product, division term to term

' transpose

[] get a value element

All this operation are done at compilation time, so when you write

```
([dx(u),dy(u)]*[dx(v),dy(v)]')[0][0] )
```

So this is completly equivalent to with no extra computation at exec time

```
dx(u)*dx(v)
```

Init Problem

`FreeFem++` reads a user's init file named `freefem++.pref` to initialize global variables : `verbosity`, `includepath`, `loadpath`.

Remark : the variable `verbosity` changes the level of internal printing (0, nothing (except mistake), 1 few, 10 lots, etc. ...), the default value is 2. The include files are searched from the `includepath` list and the load files are searched from `loadpath` list.

The syntax of the file is :

```
verbosity= 5
loadpath += "/Library/FreeFem++/lib"
loadpath += "/Users/hecht/Library/FreeFem++/lib"
includepath  += "/Library/FreeFem++/edp"
includepath  += "/Users/hecht/Library/FreeFem++/edp"
#  comment
```

```
load += "funcTemplate"
load += "myfunction"
```

The possible paths for this file are

– under unix and MacOs

```
/etc/freefem++.pref
   $(HOME)/.freefem++.pref
   freefem++.pref
```

– under windows

```
freefem++.pref
```

We can also use shell environment variable to change verbosity and the search rule before the init files.

```
export FF_VERBOSITY=50
export FF_INCLUDEPATH="dir;;dir2"
export FF_LOADPATH="dir;;dir3""
```

Remark : the separator between directories must be ” ;” and not ” :” because ” :” is used under Windows.

Remark, to show the list of init of `freefem++`, do

```
Brochet$ export FF_VERBOSITY=100; ./FreeFem++-nw
 --  verbosity is set to 100
 insert init-files /etc/freefem++.pref $
...
```

load facily problem, find

```
load("msh3") ;
```

try to load file `msh3.so`,`msh3.dll` ,`msh3.dylib` (depend of OS)

To see the initial process :

```
Brochet-2:~ hecht$ (export FF_VERBOSITY=1000;FreeFem++)
 insert init-files /etc/freefem++.pref $
 insert init-files /Applications/FreeFem++.app/Contents//etc/freefem++.pref $
 insert init-files /usr/local/lib/ff++/3.5/etc/freefem++.pref $
 insert init-files /Users/hecht/.freefem++.pref $
 insert init-files freefem++.pref $
 try initfile : /etc/freefem++.pref
error opening init  file: /etc/freefem++.pref
 try initfile : /Applications/FreeFem++.app/Contents//etc/freefem++.pref
/Applications/FreeFem++.app/Contents//etc/freefem++.pref:loadpath = /usr/local/lib/ff++/3.0
 insert loadpath /usr/local/lib/ff++/3.0-4/lib/ $
/Applications/FreeFem++.app/Contents//etc/freefem++.pref:includepath = /usr/local/lib/ff++/
 insert includepath /usr/local/lib/ff++/3.0-4/edp/ $
 try initfile : /usr/local/lib/ff++/3.5/etc/freefem++.pref
/usr/local/lib/ff++/3.5/etc/freefem++.pref:loadpath = ./
 insert loadpath ./ $
```

```
/usr/local/lib/ff++/3.5/etc/freefem++.pref:loadpath = /usr/local/lib/ff++/3.5/lib
 insert loadpath /usr/local/lib/ff++/3.5/lib/ $
/usr/local/lib/ff++/3.5/etc/freefem++.pref:includepath = /usr/local/lib/ff++/3.5/edp
 insert includepath /usr/local/lib/ff++/3.5/edp/ $
 try initfile : /Users/hecht/.freefem++.pref
/Users/hecht/.freefem++.pref:loadpath = ./
 insert loadpath ./ $
/Users/hecht/.freefem++.pref:loadpath = /Users/hecht/Library/FreeFem++/lib
 insert loadpath /Users/hecht/Library/FreeFem++/lib/ $
/Users/hecht/.freefem++.pref:includepath = /Users/hecht/Library/FreeFem++/edp
 insert includepath /Users/hecht/Library/FreeFem++/edp/ $
 try initfile : freefem++.pref
error opening init  file: freefem++.pref

load path :
/usr/local/lib/ff++/3.0-4/lib/
/usr/local/lib/ff++/3.5/lib/
./
/Users/hecht/Library/FreeFem++/lib/
(.)

include path :
/usr/local/lib/ff++/3.0-4/edp/
/usr/local/lib/ff++/3.5/edp/
/Users/hecht/Library/FreeFem++/edp/
(.)
```

```
--  verbosity is set to 1000
EXEC of the plot  : ffglut
Syntaxe = FreeFem++ [ -v verbosity ] [ -fglut filepath ] [ -glut command ] [ -nw] [ -f] fi
       -v      verbosity :  0 -- 1000000 level of freefem output
       -fglut  filepath  :  the file name of save all plots (replot with ffglut command )
       -glut    command  :  the command name of ffglut
       -nowait           : nowait at the end on window
       -wait             : wait at the end on window
       -nw               :  no ffglut (=> no graphics windows)
with        default ffglut : ffglut
```