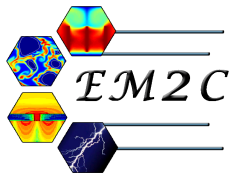


# VISUALISATION AVEC GNU PLOT

Fabien THOLIN

October 14, 2011



Intro

Expression  
analytiques

Plots partir  
d'un fichier  
Multiplot

SCRIPTS

Scripts  
avancés

Conclusion

- Logiciel libre de traçage graphique en lignes de commande
- Logiciel maintenu et en constante amélioration depuis 1986
- Copyright (C) 1986, Thomas Williams, Colin Kelley
- Utilisation interactive ou par scripts
- Plots 1D, 2D, 3D ?, Multiplots
- Linux, Windows, Mac OS X

```
[fabient@anila figs]$ gnuplot
```

```
GNUPLOT
```

```
Version 4.4 patchlevel 0
```

```
last modified March 2010
```

```
System: Linux 2.6.40.4-5.fc15.x86_64
```

Copyright (C) 1986-1993, 1998, 2004, 2007-2010

Thomas Williams, Colin Kelley and many others

gnuplot home: <http://www.gnuplot.info>

faq, bugs, etc: type "help seeking-assistance"

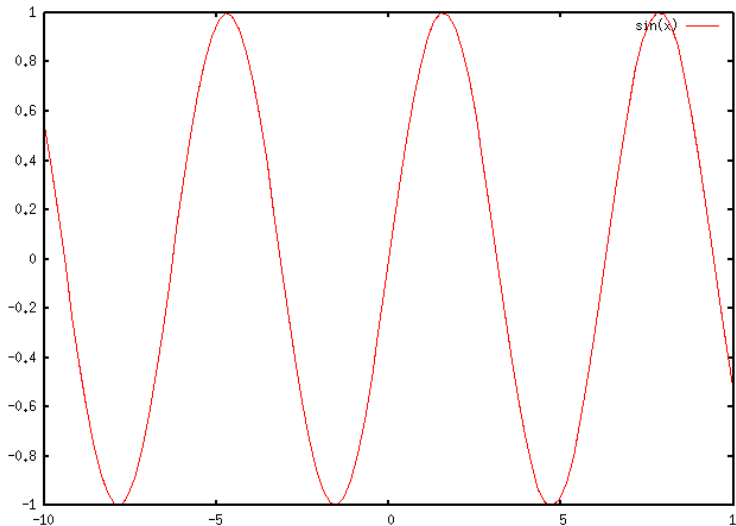
immediate help: type "help"

plot window: hit 'h'

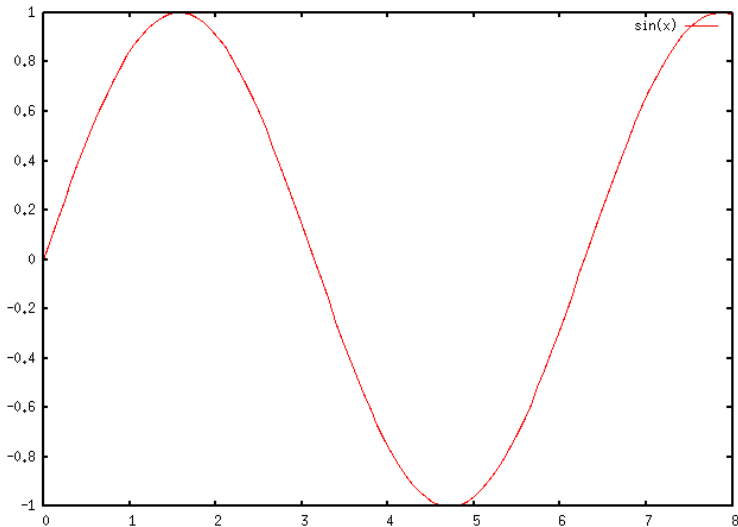
Terminal type set to 'x11'

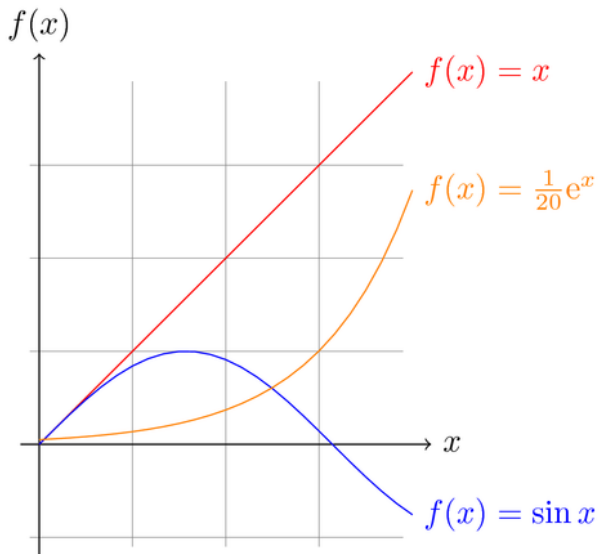
```
gnuplot>
```

```
> plot sin(x)
```



```
> set xrange[0:8]  
> plot sin(x)
```





Intro

Expression  
analytiquesPlots partir  
d'un fichier

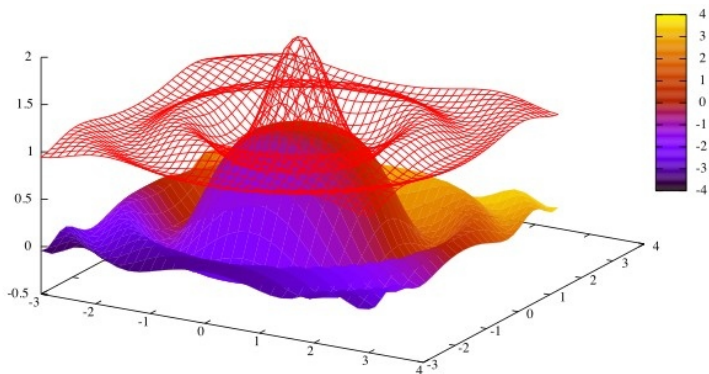
Multiplot

SCRIPTS

Scripts  
avancés

Conclusion

```
> plot sin(sqrt(x**2+y**2))/sqrt(x**2+y**2)
```



Intro

Expression  
analytiquesPlots partir  
d'un fichier

Multiplot

SCRIPTS

Scripts  
avancés

Conclusion

> **expr1 ? expr2 : expr3**

Intro

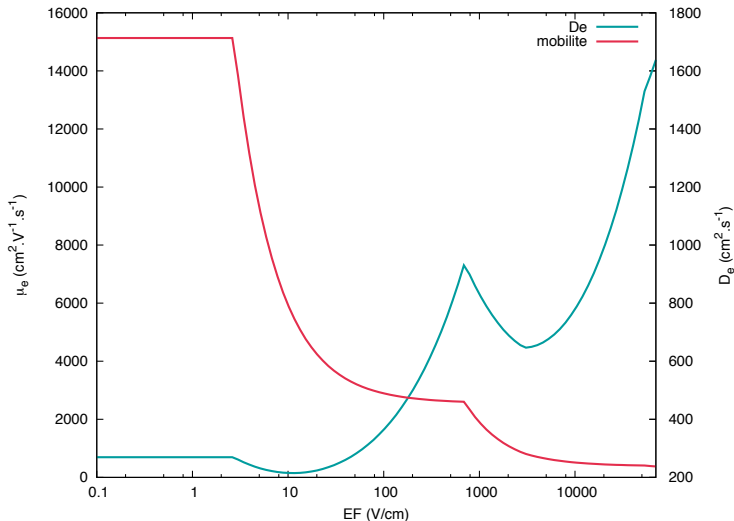
Expression  
analytiquesPlots partir  
d'un fichier

Multiplot

SCRIPTS

Scripts  
avancés

Conclusion





```
> expr1 ? expr2 : expr3
```

**intervalles :**

```
set xrange [1e-19:1e-14]
```

```
set yrange [:1000000]
```

**Définitions des fonctions :**

```
amu1(x)=(7.4e21*(x)+7.1e6)/(x*den)
```

```
amu2(x)=(1.03e22*(x)+1.3e6)/(x*den)
```

```
amu3(x)=(7.2973e21*(x)+1.63e6)/(x*den)
```

```
amu4(x)=(6.87e22*(x)+3.38e4)/(x*den)
```

```
amu5(x)=(6.87e22*(1.0e-19)+3.38e4)/(1.e-19*den)
```

**Utilisation de la fonction "ternary operator":**

```
amu(x) = ( x > 2.0e-15 ? amu1(x) : ( x > 1e-16 ? amu2(x) : ( x  
> 2.6e-17 ? amu3(x) : x > 1.e-19 ? amu4(x) : amu5(x) )))
```

**Plot :**

```
plot amu(x/den) title 'mobilite' ls 5 axis x1y1
```

**Format should be :**

**X1 Y1 F(X1,Y1)**

**X2 Y1 F(X2,Y1)**

**X3 Y1 F(X3,Y1)**

**... ... ...**

**Xn Yn F(Xn,Yn)**

**X1 Y2 F(X1,Y2)**

**X2 Y2 F(X2,Y2)**

**X3 Y2 F(X3,Y2)**

**... ... ...**

**Xn Yn F(Xn,Yn)**

**...**

Intro

Expression  
analytiquesPlots partir  
d'un fichier

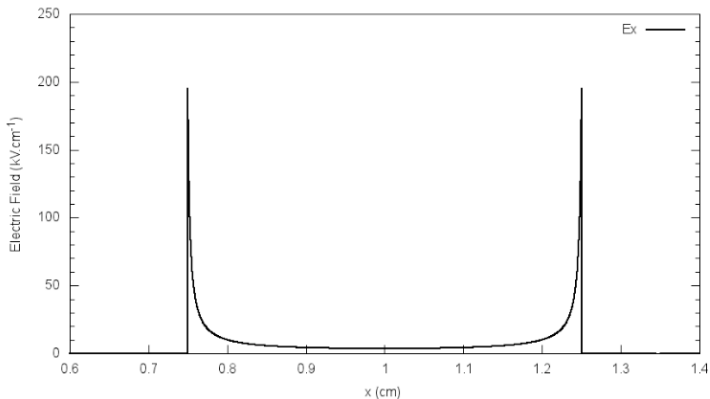
Multiplot

SCRIPTS

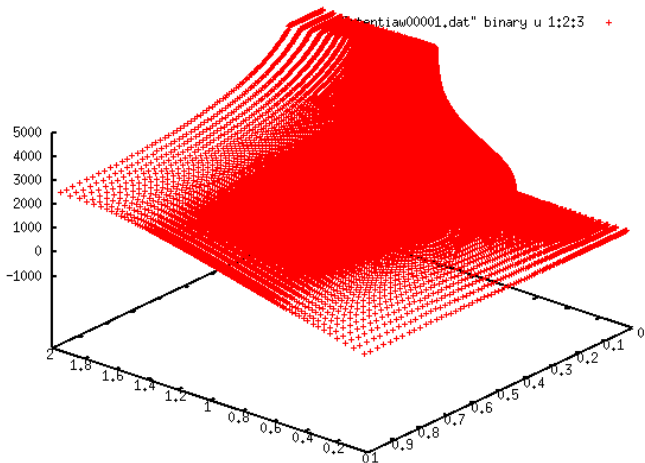
Scripts  
avancés

Conclusion

```
> plot "ddaxe_exw00001.dat" u 1:(-0.001*$ 2) lt -1 lw 2  
title "Ex" axis x1y1
```



```
splot "Potentiaw00001.dat" binary u 1:2:(0.001*$3)
```



Intro

Expression  
analytiquesPlots partir  
d'un fichier

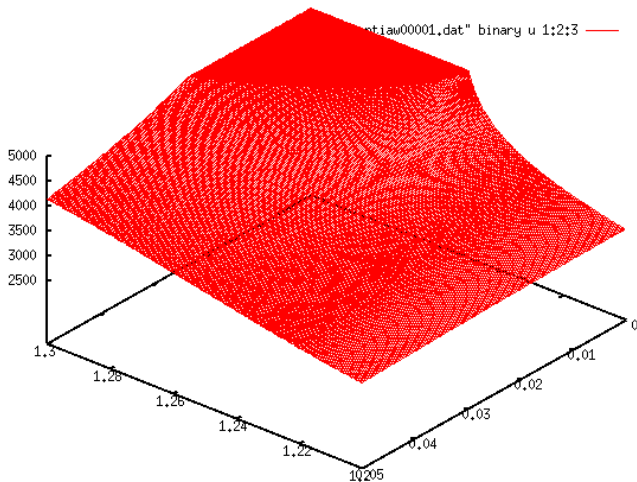
Multiplot

SCRIPTS

Scripts  
avancés

Conclusion

```
splot "Potentiaw00001.dat" binary u 1:2:(0.001*$3)
```



Intro

Expression  
analytiquesPlots partir  
d'un fichier

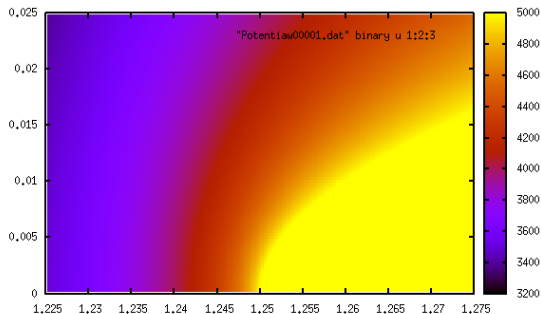
Multiplot

SCRIPTS

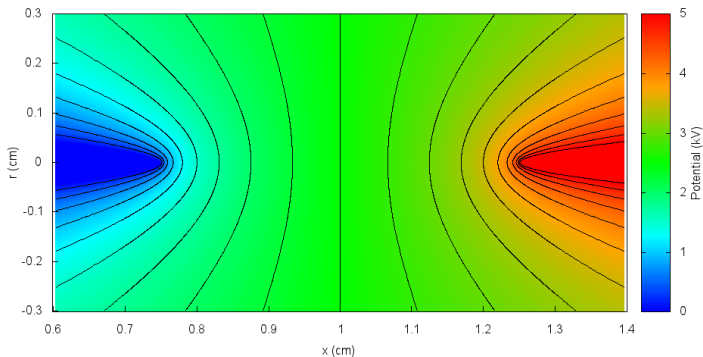
Scripts  
avancés

Conclusion

```
set pm3d  
unset surf  
set view map  
splot "Potentiaw00001.dat" binary u 1:2:(0.001*$3)
```

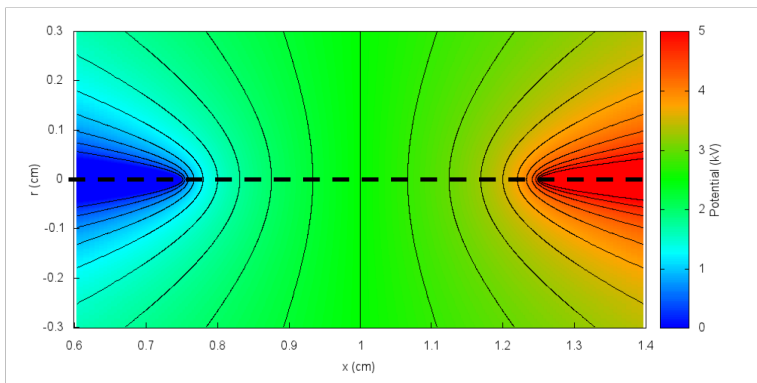


```
set pm3d; unset surf; set view map
set palette defined ( 0 'blue', 250 'cyan', 500 'green',750
'orange', 1000 'red')
set autoscale
set contour base; set cntrparam levels incremental 0,0.250,5
splot "Potentiaw00001.dat" binary u 1:2:(0.001*$3) , \
"" binary u 1:(-$2):(0.001*$3)
```



## Plots partir d'un fichier : coupes

...  
**splot "Potentiaw00001.dat" binary u 1:2:(0.001\*\$3) lt -1,\  
"" binary u 1:(-\$2):(0.001\*\$3) lt -1**



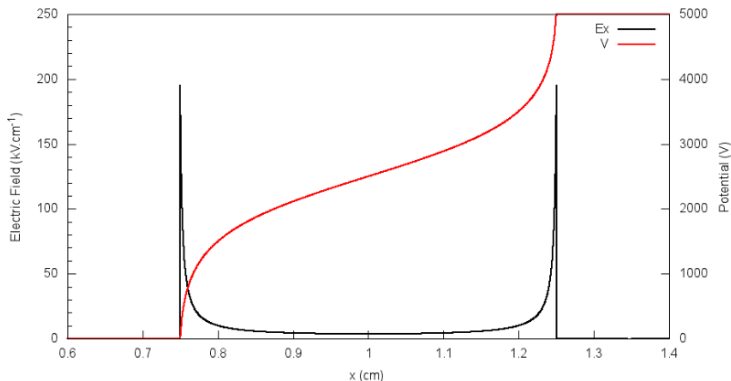


On utilise "every" pour faire des coupes :

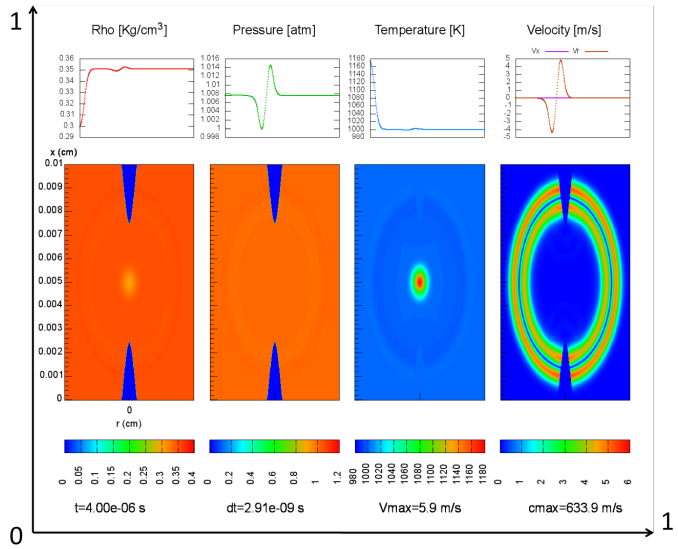
plot ... every  $\overbrace{\text{Ligne : Block}}^{\text{incrément}} : \overbrace{\text{Ligne : Block}}^{\text{premier}} : \overbrace{\text{Ligne : Block}}^{\text{dernier}}$

- Pour faire une coupe selon  $x$  à  $r = n$  :  
plot "fichier.txt" binary u 1:3 every :::n::n
- Pour faire une coupe selon  $r$  à  $x = m$  :  
plot "fichier.txt" binary u 2:3 every ::m::m
- Pour faire une coupe en "diagonale" :  
Beaucoup plus compliqué !!!!!

```
plot "ddaxe_exw00001.dat" u 1:2 lt -1 title "Ex" axis x1y1,\  
"Potentiaw00001.dat" binary u 1:3 every :::1::1 lt 1 title "V"  
axis x1y2
```



## The "screen" coordinate system :

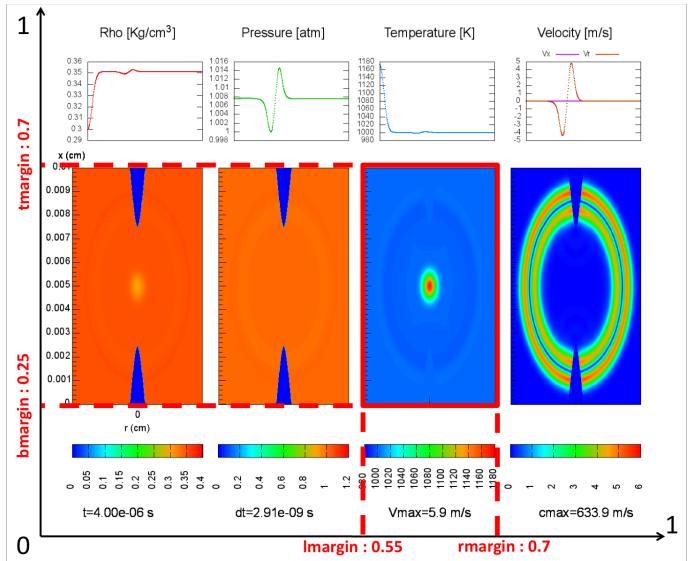


- Intro
- Expression analytiques
- Plots partir d'un fichier
- Multiplot
- SCRIPTS
- Scripts avancés
- Conclusion

Fabien  
THOLIN

## The "screen" coordinate system :

- Intro
- Expression analytiques
- Plots partir d'un fichier
- Multiplot
- SCRIPTS
- Scripts avancés
- Conclusion



**environnement "multiplot" :**

**set multiplot**

```
set lmargin screen 0.09  
set rmargin screen 0.85  
set tmargin screen 0.94  
set bmargin screen 0.765
```

**splot...**

```
set tmargin screen 0.705  
set bmargin screen 0.53
```

**splot...**

```
set tmargin screen 0.47  
set bmargin screen 0.06
```

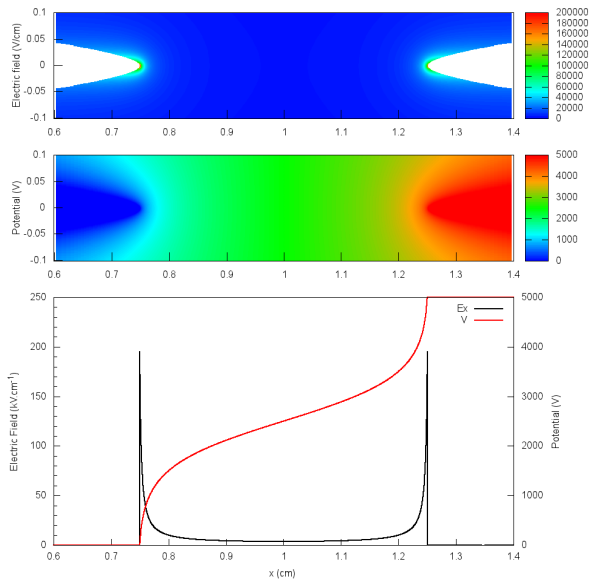
**unset colorbox**

**plot...**

Fabien  
THOLIN

## MULTIPLY :

- Intro
- Expression analytiques
- Plots partir d'un fichier
- Multiplot
- SCRIPTS
- Scripts avancés
- Conclusion



```
set term png enhanced size 1000, 1000 font "arial,20"
```

- **utilisation de chaines de caractères :**

```
FILE="SD1_sym"
```

...

```
splot FILE."/dd2DEtotw00001.dat" binary u 1:2:3,\  
"" binary u 1:(-$2):3
```

- **utilisation de "macros" :**

```
set macros  
police=" 'arial,25' "
```

...

```
set xtics font @police  
set xlabel "r (cm)" font @police offset 0,0.5,0
```

- **utilisation de "paramètres" :**

```
set lmargin screen lm  
set rmargin screen rm  
set bmargin screen bm  
set tmargin screen tm
```

## Comment faire une boucle pour appliquer le script sur un grand nombre de fichiers ?

- fichier principal : définition de fonctions  
`outputfile(x)=sprintf("out%05d.png", x)`  
`EF(x)= sprintf("../SD1/dd2DEtotw%05d.dat", x)`  
`set term png enhanced size 1500, 1000 font "arial,20"`  
`i=1`  
`load "looper"`
- fichier "Looper" : utilisation de "reread"  
`set output outputfile(i)`  
`set multiplot`  
`...`  
`plot EF(i) binary u 2:(0.001*$3)`  
`...`  
`show output`  
`i=i+1`  
`if(i < 100) reread`



## Comment lire des données dans un fichier ?

- fichier principal : utilisation de fonctions **shell** dans **Gnuplot**

```
VAR_FILE(x,y)=sprintf(" ! sed -n '%dp' ../Toutput.dat |  
awk '{printf \"%%6.2e\", $%d}' ",x,y)
```

```
... set term png enhanced size 1500, 1000 font "arial,20"  
i=1
```

```
load "looper"
```

- fichier "Looper" : utilisation de "reread"

```
...  
time=system(VAR_FILE(i,1)) ...
```

```
show output
```

```
i=i+1
```

```
if(i < 100) reread
```

**lancer gnuplot en parallele à partir d un script écrit en shell**

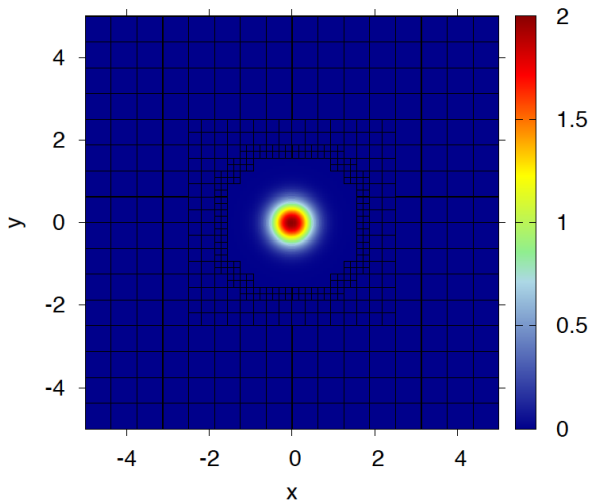
**Choix du nombre de processeurs et fonction "wait" :**

```
MAXPROC=4
NPROC=0
function check_wait(){
NPROC=${NPROC+1}
echo "Nproc=" $NPROC
if [ $NPROC == $MAXPROC ]; then
NPROC=0
echo "waiting..."
wait
fi
}
i=0
```

**debut de la boucle sur les noms de fichier :**

- **debut de la boucle sur les noms de fichier :**  
`for file in ../OUT/dd2Delecw* ; do`  
`i=$(( $i + 1 ))`
- **nom de fichier et entier à incrémenter :** `iii='printf`  
`"%05d" $i'`  
`ee_file='printf "ee%03d.png" $i'`
- **lecture de données dans un fichier :**  
`time='sed -e "$i!d" Toutput.dat | awk '{print $1}'"`
- **script gnuplot entre les BALISES "Here-Document"**  
`cat > script_gnuplot$iii << BALISE`  
`SCRIPT GNUPLOT: splot "../OUT/file$iii.dat" u`  
`1:2:(-\$3)`  
`BALISE`
- **lancement du script en parallele :**  
`gnuplot script_gnuplot$iii & check_wait`  
**fin de la boucle**

## AMR : X | Y | level | EF



Intro

Expression  
analytiquesPlots partir  
d'un fichier

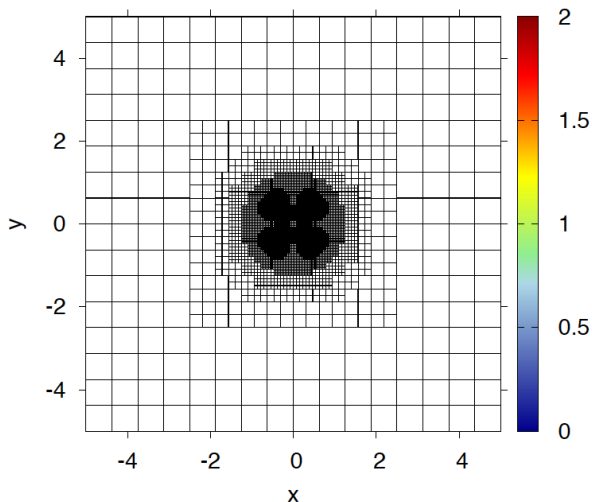
Multiplot

SCRIPTS

Scripts  
avancés

Conclusion

## AMR : X | Y | level | EF



Intro

Expression  
analytiquesPlots partir  
d'un fichier

Multiplot

SCRIPTS

Scripts  
avancés

Conclusion

- **Fonction pour créer un rectangle de centre a,b**  
`set_object_faces(a,b,c,d) = ( $c$ > threshold_face ?  
 sprintf(" ") \  
 : sprintf("set object rect center first %g , first %g\  
 size first dx_l(%g),\  
 first dx_l(%g) fs empty\  
 border rgb \"black\" lw 1.0;\n", a,b,c,c,d))`
- **fonction de concatenation :**  
`FACES=""  
 face(a,b,c,d)= ( FACES =  
 FACES.set_object_faces(a,b,c,d), a,b,c,d)`
- **Utilisation de plot pour remplir la chaine "FACES"**  
`set table "a"  
 plot "Champ2d-gd.dat" u 1:(face(1,2,3,4))  
 unset table "a"`
- **Tracé du maillage en "exécutant" la chaine "FACES"**  
`eval(FACES)`

Intro

Expression  
analytiquesPlots partir  
d'un fichier

Multiplot

SCRIPTS

Scripts  
avancés

Conclusion

## CONCLUSION :

- Gnuplot est très performant pour les sorties 1D et 2D
- Il permet de réaliser des scripts relativement simples et complets
- Il autorise une utilisation plus complexe :
  - création automatique d'outputs en parallèles
  - utilisation de Gnuplot et du shell
  - définition de fonctions
  - plots sur maillages AMR / Non structurés