



# Improving figures using TikZ/PGF for LATEX: An Introduction

Vincent Darrigrand, Florian Faucher

## ► To cite this version:

Vincent Darrigrand, Florian Faucher. Improving figures using TikZ/PGF for LATEX: An Introduction. Doctoral. Bilbao, Spain. 2016. cel-01400571

**HAL Id: cel-01400571**

**<https://hal.science/cel-01400571>**

Submitted on 22 Nov 2016

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Improving figures using TikZ/PGF for L<sup>A</sup>T<sub>E</sub>X: An Introduction

Vincent Darrigrand <sup>1,2</sup>    Florian Faucher <sup>2</sup>

<sup>1</sup>University of the Basque Country (UPV/EHU), Leioa, Spain,

<sup>2</sup>INRIA - MAGIQUE 3D, University of Pau (UPPA), France,

May 25, 2016  
BCAM  
Bilbao, Spain

- Examples and “short” tutorials online.
  - [en.wikipedia.org/wiki/PGF/TikZ](http://en.wikipedia.org/wiki/PGF/TikZ)
  - [www.texample.net/tikz/resources/](http://www.texample.net/tikz/resources/)
  - [www.texample.net/tikz/examples/](http://www.texample.net/tikz/examples/)
- List of packages TikZ/PGF:
  - [www.ctan.org/topic/pgf-tikz](http://www.ctan.org/topic/pgf-tikz)
- PGF Manual:
  - [mirrors.ctan.org/graphics/pgf/base/doc/pgfmanual.pdf](http://mirrors.ctan.org/graphics/pgf/base/doc/pgfmanual.pdf)
- Pgfplots manual:
  - [mirrors.ctan.org/graphics/pgf/contrib/pgfplots/doc/pgfplots.pdf](http://mirrors.ctan.org/graphics/pgf/contrib/pgfplots/doc/pgfplots.pdf)

# Learning process

- 1 Working from already existing files.
- 2 Generation of TiKZ examples from software: Geogebra, Matlab, R, inkscape...
- 3 Understand and adapt the examples.
- 4 Make your own pictures

# Introduction

```
\usepackage{pgfplots} % loads TiKZ/PGF and pgfplots  
\usepgfplotslibrary{library_name}  
\usetikzlibrary{library_name}
```

```
\begin{figure}  
  \begin{tikzpicture}
```

⋮

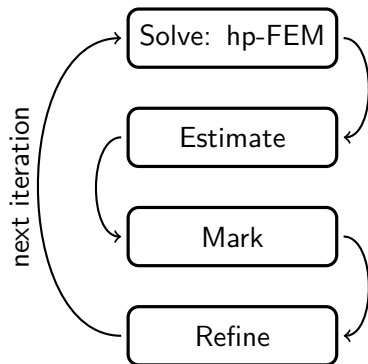
*Here is the code for your picture*

⋮

```
  \end{tikzpicture}  
  \caption{my caption}  
\end{figure}
```

- 1 Algorithm
- 2 Computational domain sketch
- 3 Data plot

# Algorithm



**Figure:** Algorithm related graph



```
\node[boxOptions]  
(sol){Solve: hp-FEM};
```

Solve: hp-FEM

```
\node[boxOptions]  
(sol){Solve: hp-FEM};
```

Solve: hp-FEM

```
\tikzset{boxOptions/.style={  
rectangle,  
rounded corners,  
draw=black, very thick,  
text width=6.5em,  
minimum height=2em,  
text centered}  
}
```

```
\node[boxOptions]  
(sol){Solve: hp-FEM};
```

Solve: hp-FEM

```
\node[boxOptions,below=of  
sol]  
(estim){Estimate};
```

Estimate

```
\node[boxOptions]  
(sol){Solve: hp-FEM};
```

Solve: hp-FEM

```
\node[boxOptions,below=of  
sol]  
(estim){Estimate};
```

Estimate

```
\node[boxOptions,below=of  
estim]  
(mark){Mark};
```

Mark

```
\node[boxOptions]  
(sol){Solve: hp-FEM};
```

Solve: hp-FEM

```
\node[boxOptions,below=of  
sol]  
(estim){Estimate};
```

Estimate

```
\node[boxOptions,below=of  
estim]  
(mark){Mark};
```

Mark

```
\node[boxOptions,below=of  
mark]  
(ref)Refine;
```

Refine

```
\draw[arrowStyle]
(sol.east) to[out=0,in=0]
(estim.east);

\tikzset{arrowStyle/.style={
->,
thick,
shorten <=2pt,
shorten >=2pt},
```

Solve: hp-FEM

Estimate

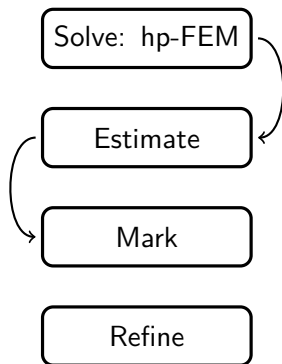
Mark

Refine



```
\draw[arrowStyle]  
(sol.east) to[out=0,in=0]  
(estim.east);
```

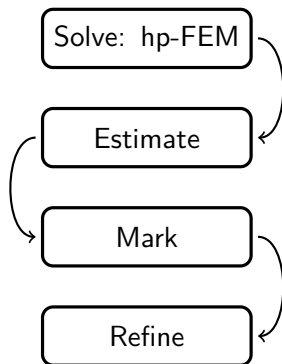
```
\draw[arrowStyle]  
(estim.west) to[out=180,  
in=180] (mark.west);
```



```
\draw[arrowStyle]
(sol.east) to[out=0,in=0]
(estim.east);

\draw[arrowStyle]
(estim.west) to[out=180,
in=180] (mark.west);

\draw[arrowStyle]
(mark.east) to[out=0,in=0]
(ref.east);
```





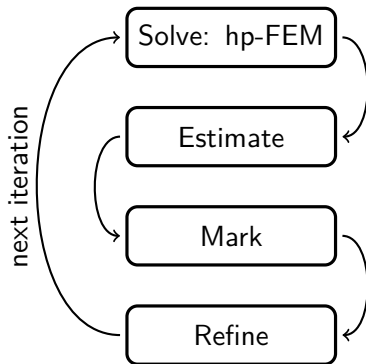
# Algorithm

```
\draw[arrowStyle]  
(sol.east) to[out=0,in=0]  
(estim.east);
```

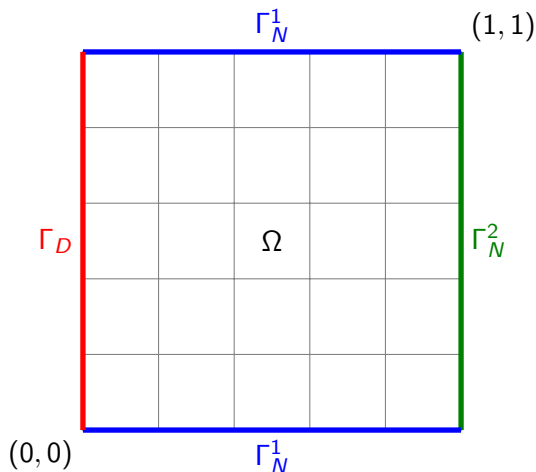
```
\draw[arrowStyle]  
(estim.west) to[out=180,  
in=180] (mark.west);
```

```
\draw[arrowStyle]  
(mark.east) to[out=0,in=0]  
(ref.east);
```

```
\draw[arrowStyle]  
(ref.west) to[out=180,in=180]  
node[sloped, anchor=south]  
{next iteration}(sol.west);
```



## Computational domain sketch

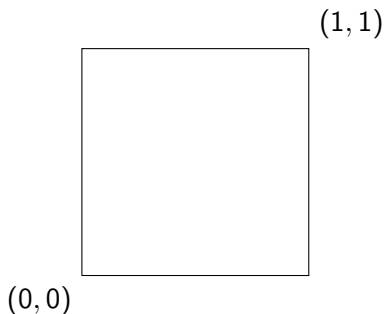


**Figure:** 2D Computational domain

## Computational Domain: Step by step

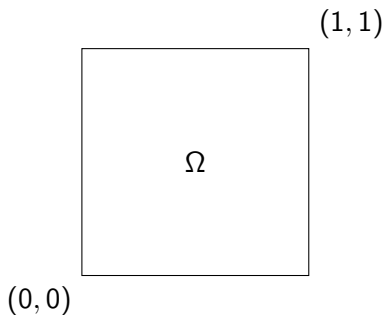
```
\draw[color=black] (0,0)  
rectangle (1,1);
```

```
\node[anchor=north east]  
at (0,0){(0,0)};  
\node[anchor=south west]  
at (1,1){(1,1)};
```



# Computational Domain: Step by step

```
\draw[color=black] (0,0)  
rectangle (1,1);  
\node at (0.5,0.5) {\Omega};
```



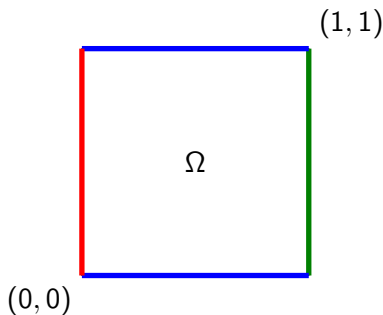
# Computational Domain: Step by step

```
\draw[border_style,  
color=green!50!black] (1,0)  
- (1,1);  
\draw[border_style,  
color=blue] (0,1) - (1,1);
```

```
\draw[border_style,  
color=blue] (0,0) - (1,0);
```

```
\draw[border_style,  
color=red,] (0,0) - (0,1);
```

```
\tikzset{border_style/.style  
= {line width=2pt}}
```



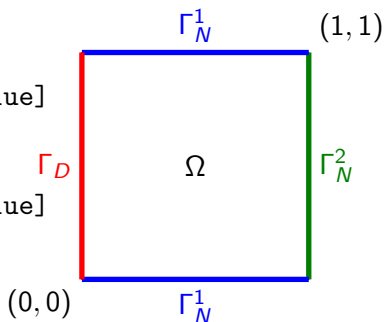
# Computational Domain: Step by step

```
\node[anchor=east, color=red]  
at (0,1/2) {\Gamma_D};
```

```
\node[anchor=south, color=blue]  
at (1/2,1) {\Gamma_N^1};
```

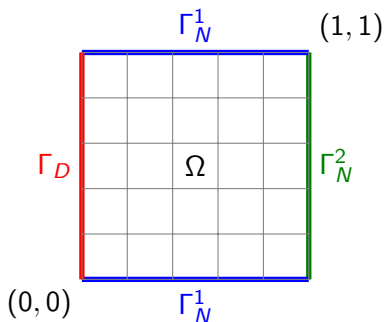
```
\node[anchor=north, color=blue]  
at (1/2,0) {\Gamma_N^1};
```

```
\node[anchor=west,  
color=green!50!black] at  
(1,1/2) {\Gamma_N^2};
```



# Computational Domain: Step by step

```
\draw[step=0.2,gray]  
(0,0) grid (1,1);
```





**Data plot**

| epsilon              | error1               | error2               |
|----------------------|----------------------|----------------------|
| $7.1 \cdot 10^{-3}$  | $1.51 \cdot 10^{-2}$ | $2.27 \cdot 10^{-3}$ |
| $9.5 \cdot 10^{-3}$  | $2.01 \cdot 10^{-2}$ | $3.96 \cdot 10^{-3}$ |
| $1.28 \cdot 10^{-2}$ | $2.69 \cdot 10^{-2}$ | $5.84 \cdot 10^{-3}$ |
| $1.73 \cdot 10^{-2}$ | $3.6 \cdot 10^{-2}$  | $1.23 \cdot 10^{-2}$ |
| $2.32 \cdot 10^{-2}$ | $4.76 \cdot 10^{-2}$ | $2.61 \cdot 10^{-2}$ |
| $3.12 \cdot 10^{-2}$ | $6.28 \cdot 10^{-2}$ | $2.51 \cdot 10^{-2}$ |

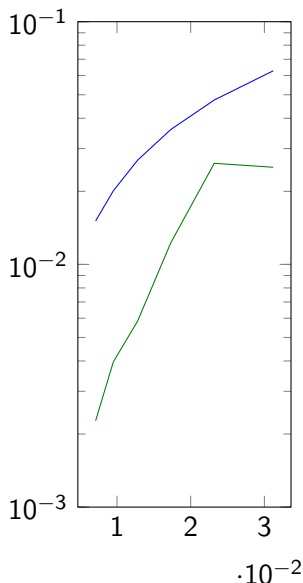
# Data Plots: a simple example

```
\begin{axis}[axis options]
```

```
\addplot[color=blue]  
table[x=epsilon,y=error1]  
{./data.txt};
```

```
\addplot[color=green!50!black]  
table[x=epsilon,y=error2]  
{./data.txt};
```

```
\end{axis}
```



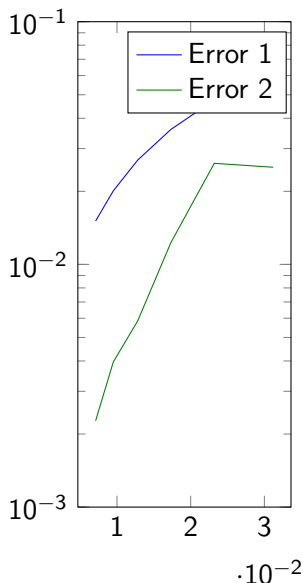
# Data Plots: a simple example

```
\begin{axis}[axis options]
```

```
\addplot[color=blue]  
table[x=epsilon,y=error1]  
{./data.txt};  
\addlegendentry{error 1};
```

```
\addplot[color=green!50!black]  
table[x=epsilon,y=error2]  
{./data.txt};  
\addlegendentry{error 2};
```

```
\end{axis}
```



# Data Plots: a simple example

```
\begin{axis}[axis options]
```

```
\addplot[color=blue]
```

```
table[x=epsilon,y=error1]
```

```
{./data.txt};
```

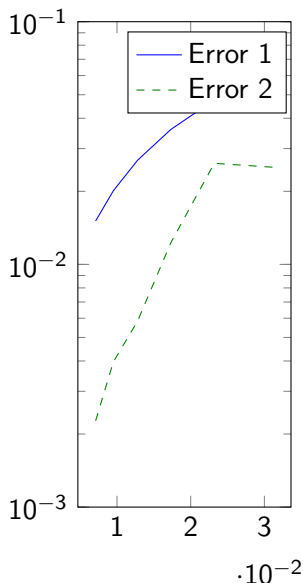
```
\addlegendentry{error 1};
```

```
\addplot[color=green!50!black,  
dashed] table[x=epsilon,y=error2]
```

```
{./data.txt};
```

```
\addlegendentry{error 2};
```

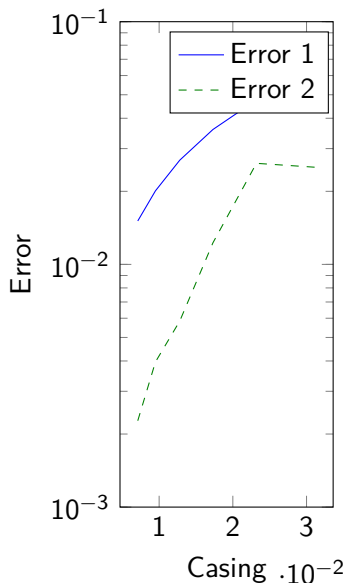
```
\end{axis}
```



# Data Plots: a simple example

*Axis options:*

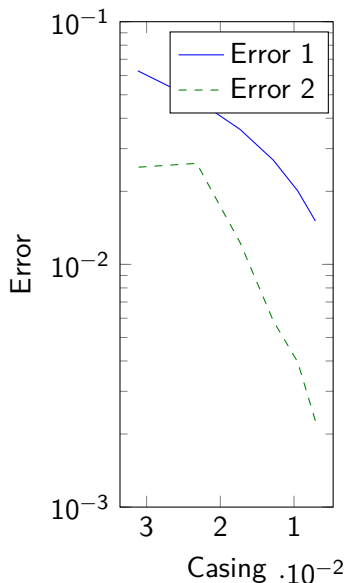
```
y mode=log,  
y min=1e-3,  
y max=1e-1,  
x label=Casing,  
y label=Error,
```



# Data Plots: a simple example

*Axis options:*

```
y mode=log,  
y min=1e-3,  
y max=1e-1,  
x label=Casing,  
y label=Error,  
x dir=reverse,
```



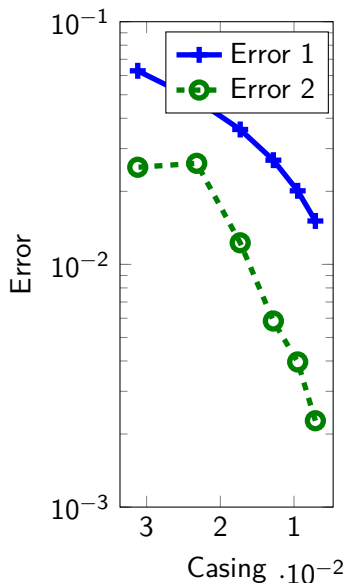
# Data Plots: a simple example

*Axis options:*

```
y mode=log,  
y min=1e-3,  
y max=1e-1,  
x label=Casing,  
y label=Error,  
x dir=reverse,
```

*More plot options:*

```
mark=+, mark=o,  
mark size=3pt,  
line width=2pt,
```

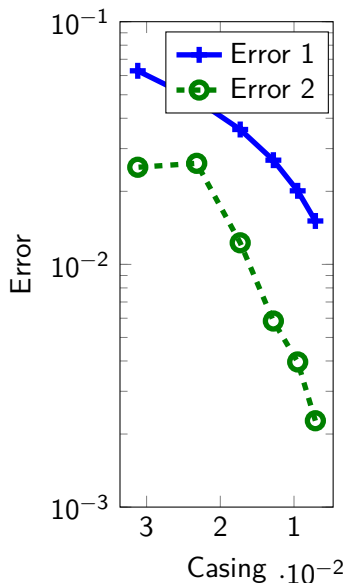




# Data Plots: a simple example

*Legend options:*

legend pos=north east,

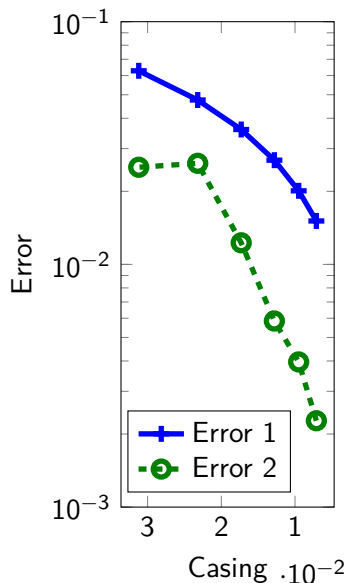


# Data Plots: a simple example

*Legend options:*

legend pos=north east,

legend pos=south west,



# Data Plots: a simple example

*Legend options:*

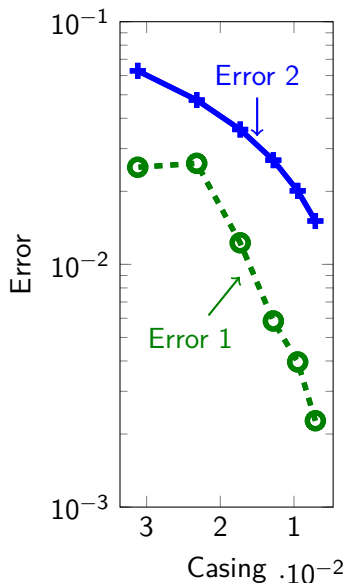
legend pos=north east,

legend pos=south west,

*Pin point: (after the  
\addplot)*

```
node[pin={ [pin edge={<-,solid,  
blue, thick}]90:Error2}
```

```
] at (1.5e-2 ,3e-2) {};
```



# Many ways and packages to improve your figures

