

---


# INTRODUCTION TO GNUPLOT FOR MESHFREE

*Plotting TimestepFile data with open-source software*

MESHFREE-Team, January 2023

---

# Command line plotting

- A TimestepFile in MESHFREE is generated in ASCII format 
  - Can be visualized by a variety of software
  - Under Linux OS, GNUPLOT is typically available

- GNUPLOT offers a quick way to plot data from a TimestepFile through shell commands:

```
gnuplot -p -e "plot 'results_default.timestep' using 1:3"
```

- takes the first column of "results\_default.timestep" as x-data and the third column as y-data
- "-p" keeps the plot window open
- "-e" executes the following GNUPLOT command
- Append "with linespoints" to plot lines between data points

# Embedding GNUPLOT in shell scripts

- GNUPLOT commands can be embedded into a shell script:

```
#!/bin/sh
timestep_dataFile='results_default.timestep'
plotCommand='using 1:3 with linespoints'
xString='x'
yString='y'
gnuplot --persist <<- EndOfPlot
    set xlabel "${xString}"
    set ylabel "${yString}"
    plot "${timestep_dataFile}" $plotCommand
EndOfPlot
```

- GNUPLOT commands enclosed by “gnuplot --persist <<- EndOfPlot” and “EndOfPlot”
- Shell variables can be used inside the GNUPLOT commands
- Provides an easier and customizable interface for multiple GNUPLOT commands

# Saving plot to file

- Change output to save plot to an image file:

```
gnuplot --persist <<- EndOfPlot
  set terminal png size 1200,900
  set output 'gnuplot.png'
  plot "${timestep_dataFile}" $plotCommand
EndOfPlot
```

- The user can still open a plot window after saving the plot to an image file:

```
gnuplot --persist <<- EndOfPlot
  set terminal png size 1200,900
  set output 'gnuplot.png'
  plot "${timestep_dataFile}" $plotCommand
  set terminal x11
  set output
  replot
EndOfPlot
```

# Example of a reusable shell script

```
#!/bin/sh
### USER INPUT: ###
SAVE_path="results"
SAVE_file="TrainingSetup"
timestep_name="default" #"default" if no timestep file name is specified
xCol=0 #Column index of x-data (starting from 0)
yCol=4 #Column index of y-data (starting from 0)
figTitle="" #title of gnuplot figure
pngFile="output_GNUPLOT" #File name of png file (without extension)
### SHELL Processing: ###
timestep_dataFile="${SAVE_path}/${SAVE_file}_${timestep_name}.timestep"
timestep_headerFile="${timestep_dataFile}.header"
mapfile -t LIST_OF_HEADERS < $timestep_headerFile
xString="${LIST_OF_HEADERS[${xCol}]}"
yString="${LIST_OF_HEADERS[${yCol}]}"
plotCommand="using $((xCol + 1)):$((yCol + 1)) with linespoints linestyle 1"
### GNUPLOT: ###
gnuplot --persist <<- EndOfPlot
  set title "${figTitle}" font ",14" textcolor rgbcolor "royalblue"
  set key off
  set xlabel "${xString}"
  set ylabel "${yString}"
  set autoscale
  set terminal png size 1200,900
  set datafile separator ","
  set output '${pngFile}.png'
  plot "${timestep_dataFile}" $plotCommand
  set terminal x11
  set output
  replot
EndOfPlot
```