

# C2110 *UNIX and programming*

## Lesson 9 / Module 1

**PS / 2020 Distance form of teaching: Rev2**

Petr Kulhanek

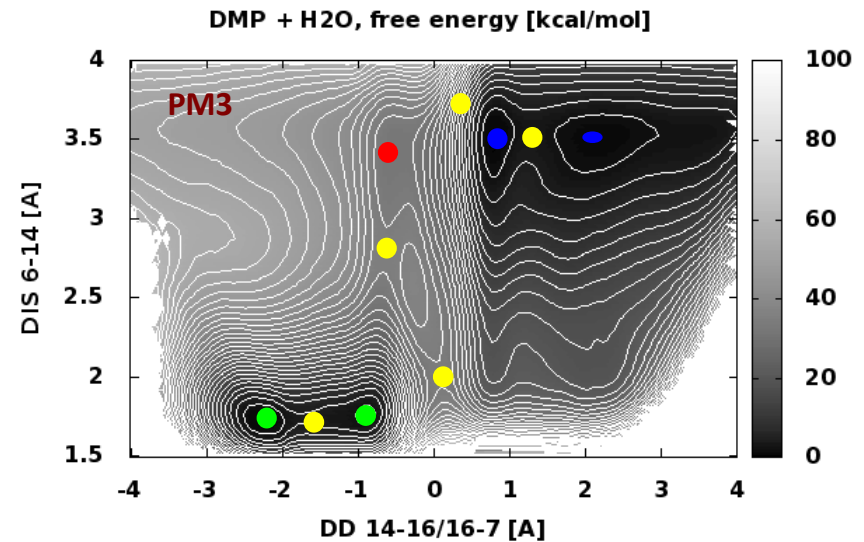
[kulhanek@chemi.muni.cz](mailto:kulhanek@chemi.muni.cz)

National Center for Biomolecular Research, Faculty of Science  
Masaryk University, Kamenice 5, CZ-62500 Brno

# Gnuplot

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

(documentation, tutorials, source codes)



# Content

## ➤ Presentation of scientific data

## ➤ Gnuplot

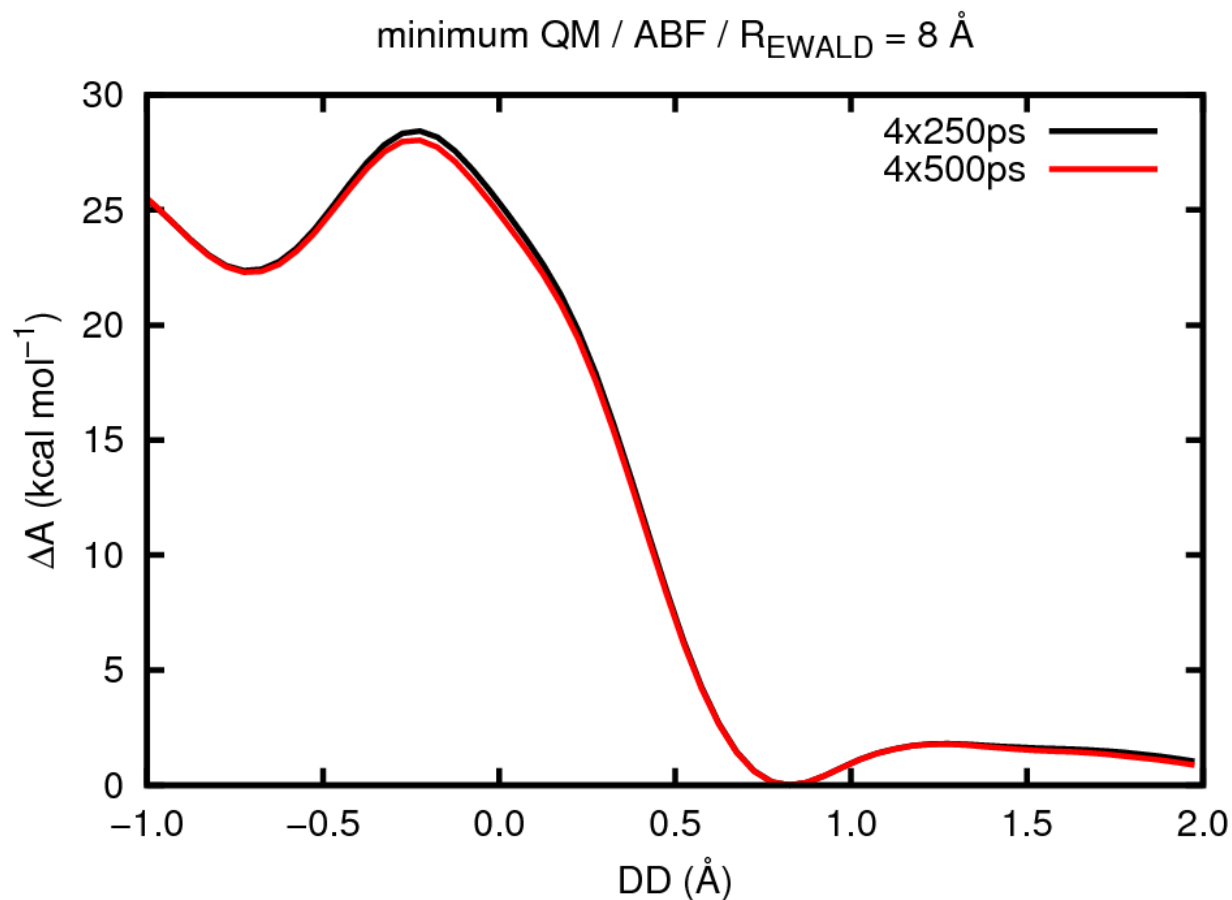
- language overview
- plot command, setting and description of axis
- terminals
- splot command

# Presentation of scientific data

## Data presented using a graph:

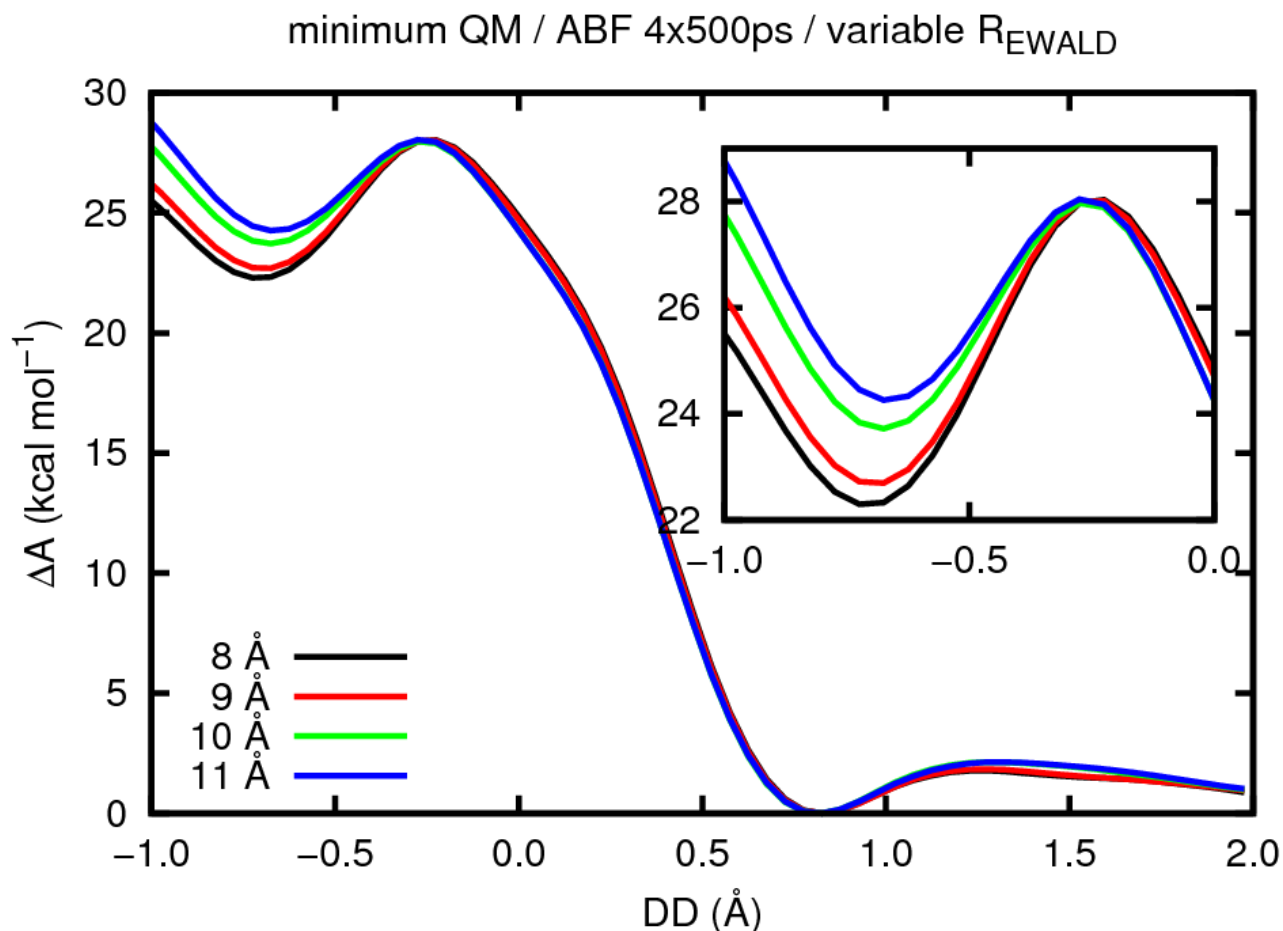
- graph must not be flooded with data
- presentation of the data must be concise and unambiguous
- axes of the graph must be described, including units
- data series must be distinguished either by color or by line style (solid, dashed, dotted, etc.) and provided with key indicated in the graph or in its key

# Samples



**Figure 1.** Impact of total simulation time on the calculated free energy  $\Delta A$  (DD).

# Samples



**Figure 2.** Impact of REWALD cut-off on the calculated free energy  $\Delta A$  ( $DD$ ). The inset shows the region with the largest deviations.

# Interactive Run

**Gnuplot** is used to draw 2D and 3D graphs and enabled work in both interactive and scripting mode.

## Interactive mode

```
[kulhanek@wolf ~]$ gnuplot
```

```
G N U P L O T
Version 4.4 patchlevel 3
last modified March 2011
System: Linux 3.2.0-31-generic
```

```
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 'wxt'
gnuplot>
```

command line of Bash shell



command line of gnuplot



# Non-interactive startup

## 1) Indirect start

We run the language interpreter and give the name of the script as an argument.

```
$ gnuplot my_gnuplot_script
```

Scripts **do not have to** have the x flag set (executable).

## 2) Direct start

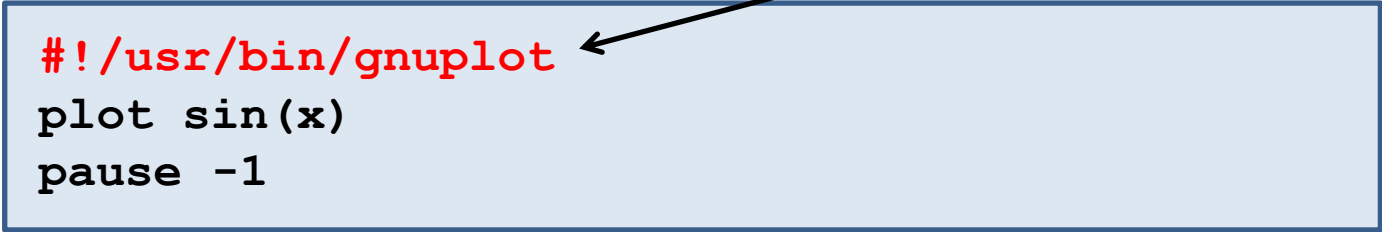
We run the script directly (shell automatically starts the interpreter).

```
$ chmod u+x my_gnuplot_script
```

```
$ ./my_gnuplot_script
```

Scripts **must** have the **x** flag (executable) and **interpreter** (part of script) set.

```
#!/usr/bin/gnuplot  
plot sin(x)  
pause -1
```





# Command - plot: functions

```
> plot function/file [display_settings] [, function/file...]
```

Displays an XY graph of a function or data series contained in a file.

## Examples (functions):

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

```
> plot cos(5.7*x+3.4)
```

```
> plot sin(x), cos(x)
```

```
> f(x) = sin(x)
```

```
> plot f(x)
```

# Command - plot: Data Series

```
> plot function/file [display_settings] [, function/file...]
```

Displays an XY graph of a function or data series contained in a file.

Examples (functions):

name of file with data (name must be enclosed in quotation marks)

```
> plot "input.txt"
```

```
> plot "input.txt", "teplota.log"
```

```
> plot "input.txt" using 2:3
```

third column forms y-values

second column forms x-values

```
> plot "input.txt" using 1: ($2+10.7)
```

```
> plot "input.txt" using ($1/10) : ($2+$3)
```

(...) an arithmetic expression at any position

\$N is value in column N

# Command – plot: Setting

```
> plot function/file [display_settings] [, function/file...]
```

Displays an XY graph of a function or data series contained in a file.

## Examples:

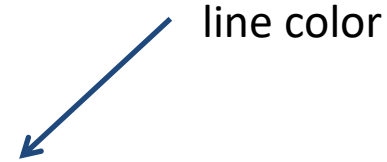
```
> plot sin(x)
```

```
> plot cos(x) title "cos"
```

```
> plot f(x) with points linecolor rgb "red" linewidth 2
```



lines, points, linespoints, dots,...



line width



```
> plot "data.log" using 2:3 title "temperature" with lines \  
    linecolor rgb "red" linewidth 2 , \  
    "data.log" using 2:4 title "pressure" with lines \  
    linecolor rgb "blue" linewidth 2
```

# Exercise I

1. Display the function  $y=x^2$
2. Display the function from the first exercise in blue
3. View the temperature profile over time contained in the file  
`/home/kulhanek/Documents/C2110/Lesson09/temp.txt`  
The time is given in the first column, the temperature is given in the second column.
4. Display the  $\sin(x)$  function with a red line and the  $\cos(x)$  function with an orange line and dots in one graph.

**Solve tasks in interactive mode.**

# Other Commands

- > `set title "popis"` # graph header (title)
- > `set xrange [min_value:max_value]` # sets range of x-axis
- > `set xlabel "description"` # sets description of x-axis
- > `set yrange [min_value:max_value]` # sets range of y-axis
- > `set ylabel "popis"` # sets description of y-axis
- > `set nokey` # hides key to data series
- > `pause -1` # waits for press of a key

# Exercise II

1. Write a script that displays function  $y=x^2$  in range 0-10 for x-value. Run the script indirectly using the interpreter gnuplot.
2. Write a script that displays the temperature over time contained in file `/home/kulhanek/Documents/C2110/Lesson09/temp.txt`  
In the graph, describe the axes, including the description of units. Time is given in picoseconds, temperature in kelvins.

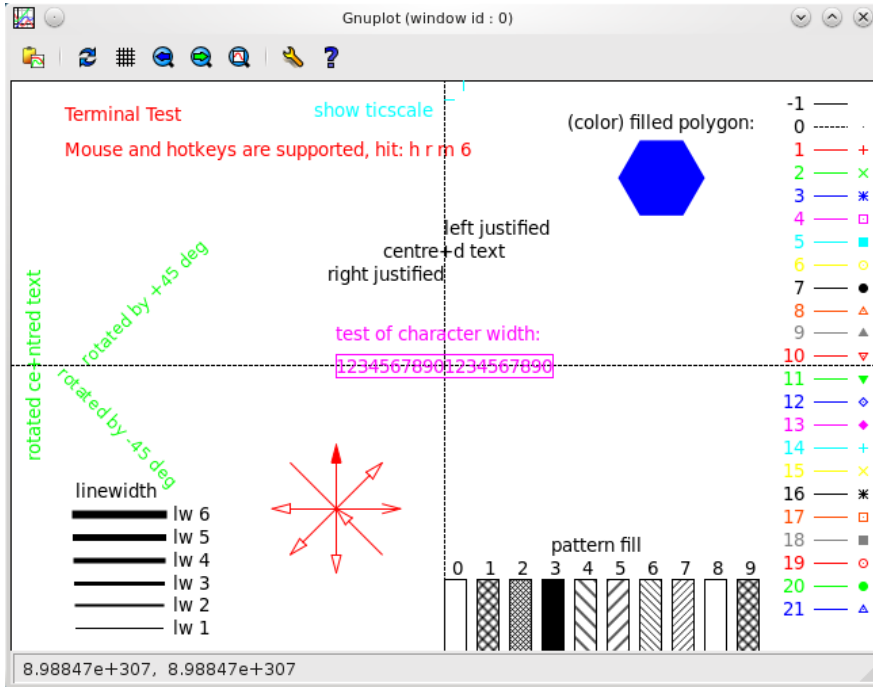
# Terminals

**Terminal** determines where the graph will be drawn.

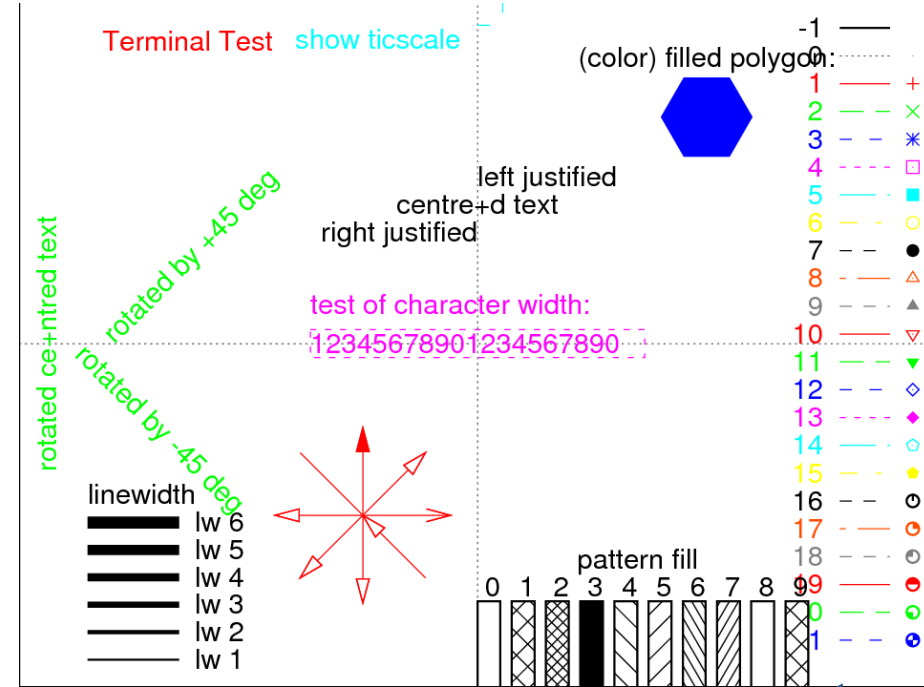
- > `set term x11` # output is plotted into a window
- > `set term qt` # output is drawn into a window (better properties)
- > `set term png size 800,600`  
# output is plotted as an image in png format
- > `set output "output.png"` # output will be saved to a file named `output.png`,  
the name must be enclosed in quotation marks
- > `test` # prints a page demonstrating the properties of the  
terminal (not all terminals have the same output  
options)

# Output from Various Terminals

wxt



postscript/eps



supports dashed lines



# Exercise III

1. What features do x11 and qt terminals provide? Work in interactive mode and use the test command.
2. Write a script that displays the function  $y=5x^3+6x^2-7$  in the range -10 to 5 for x-value. Run the script directly with the interpreter in header of the script.
3. Modify the previous script so that the graph is rendered into an image in png format. Size of the image will be 640x480. Use the display command to show the image.
4. View the result of the test command for the png and postscript terminals.
5. What terminals does gnuplot support (set terminal with no argument)?

# Self-study

---



# Command - splot

Command `splot` can be used to display the functions of two variables.

```
> plot function/file [display_settings] [, function/file ...]
```

Displays **XYZ** graph of a function or data series contained in a file.

View direction is set by command **set view a,b**, where **a** and **b** are directional angles. The top view can be set with **set view map**

When displaying functions, the sampling density for x- a y-directions can be entered with the command **set isosamples a, b**, where **a** and **b** indicate the number of samples in the given direction.

To highlight an area with a function value, **pm3d** display can be used, e.g.:

```
> splot x*x+y*y with pm3d
```

# Exercise IV

1. Display the function  $x^2 + y^2$
2. Set top view (set view)
3. Disable top view (unset view)
4. Increase the dot density to display the function (set isosamples). Use values of 10,20; 20,10 and 20,20
5. Use the pm3d visualization
6. Set top view (set view)

**Solve tasks in interactive mode.**