

# Clean export of Matlab plots

Pierre Haessig, CentraleSupélec, Rennes, February 2017

## Comparison of export methods

When creating a plot under Matlab, it is tempting to export an image (for a report) using a **screenshot**.

**Don't do this!** It yields the ugly result of Figure 1, which looks very unprofessional (e.g. gray background).

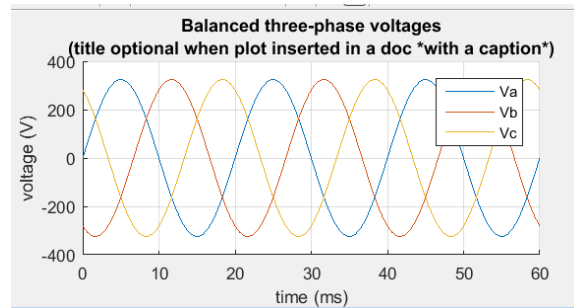


Figure 1 Dirty screenshot of the plot. Don't do this!

There are two **better alternatives**:

- use the **print** function (cf. script below).
- If you really have no time, use the "File/Save As..." menu entry of the figure window.

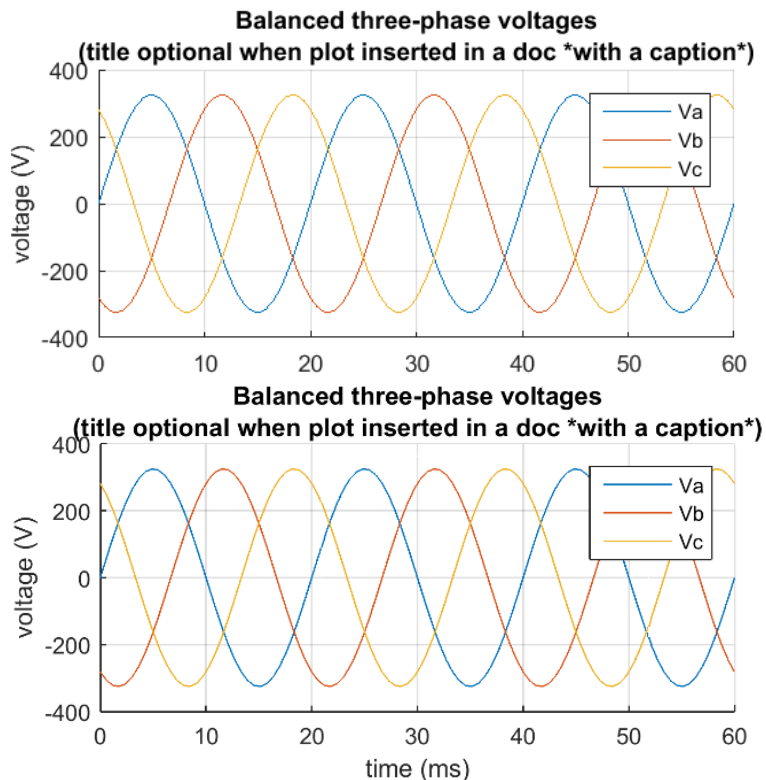


Figure 2 Clean exports using "Save As" menu (top) or the print function (bottom).  
The latter has a better resolution (smaller pixels)

You can see the result of both approaches on Figure 2. The dirty gray background is removed. Using the print function takes 30 more seconds, but you can get a much better resolution (zoom-in to compare).

### Text readability

Choose a quite **small window size** when exporting a plot image (like 30% of screen width). Otherwise, the text will look tiny on paper (cf. Figure 3).

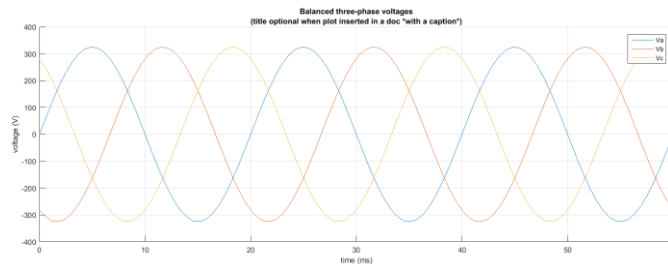


Figure 3 Export from a fullscreen plot window: text is not readable!

### Choice of image format

To export a plot, you have to choose a file format. The two most common choices for plot images are:

- **PNG**: pixel-base image format, to insert in Word, LibreOffice or Latex document. It is better than JPEG for most plots because PNG has no compression artifacts (lossless compression).
- **PDF**: vector format, to insert in Latex documents. Advantage: “infinite” resolution.

Also, Matlab can export “.fig” files which is a Matlab-specific format that can be reopened in Matlab for further edition (like changing the title). I believe that having a clean script that generates automatically the plot from the data is more useful than having a “.fig” file.

### Matlab code example

#### 1) Some data to put on the plot: three phase voltages

```
f = 50; % Hz
w = 2*pi*f;
t = linspace(0, 3/f);
V = 230*sqrt(2);
va = V*sin(w*t);
vb = V*sin(w*t-2*pi/3);
vc = V*sin(w*t-4*pi/3);
```

## 2) Plot example, with proper labeling

```
fig = figure(1);
% Set the size of the plot windows:
% Position: Location and size of figure's drawable area as [left bottom
width height]
fig.Position = [200 200 600 300];

hold on
plot(t*1e3, va)
plot(t*1e3, vb)
plot(t*1e3, vc)
grid on

xlabel('time (ms)')
ylabel('voltage (V)')
legend('Va', 'Vb', 'Vc')
title({'Balanced three-phase voltages';
      '(title optional when plot inserted in a doc *with a caption*)'})
```

## 3) Save using the `print` function (cf. `doc print`)

```
% use `fig = gcf` ("Get Current Figure") if want to print the currently
displayed figure

% To make Matlab respect the size of the plot on screen:
fig.PaperPositionMode = 'auto';
% cf. https://fr.mathworks.com/help/matlab/creating\_plots/save-figure-
at-specific-size-and-resolution.html

% PNG:
print(fig, 'plot_print.png', '-dpng', '-r300')
% '-r300' = 300 dpi to get enough pixels. 300 is a good starting point
% PDF:
print(fig, 'plot_print.pdf', '-dpdf')
```

Notice that by default the size of the PDF is a big sheet of paper. See here for a way to resize the paper: <https://fr.mathworks.com/matlabcentral/answers/12987-how-to-save-a-matlab-graphic-in-a-right-size-pdf>. Alternative under Linux: use the `pdftocrop` command.