Getting good quality graphics inside a LaTeX document

Ignas Anikevicius

July 28, 2011

1 Introduction

For publications and scientific articles, reports and thesis the most important thing is to make well looking graphics. For that there are several guidelines, which help one to get his figures look professionally, but at the same time not to overdo things as there is a limit of how good the quality has to be.

1. Use vector graphics as much as possible. Especially where there is a lot of text in the figure. But remember, if you 'convert' a jpg to eps or svg or any other vector graphics format, you will **not** gain any quality. More on this, please use Google
2. If you have to use raster graphics, please select `png` format as a better alternative where possible
3. If you use raster graphics, do not exceed the final resolution of the picture to more than 600dpi as most of the printers are printing at 300dpi or 600dpi, so anything more than that might just be wasted time while waiting the figure to be rendered. Of course if you have a good reason why you need more than 600dpi, then go ahead.
4. Have a high quality copy of your figure somewhere in your computer. This is because while converting from one format to another one can **not** improve the quality.

If you feel that you have not found enough information on the graphics usage in LaTeX, please refer to these websites:

- Floats, Figures and Captions
- Importing Graphics
- Creating Graphics

2 Inserting a simple figure

Inserting a simple figure can be as easy as:

```
\begin{figure}[H]
    \begin{center}
        \includegraphics{plot3.eps}
    \end{center}
    \caption{First plot}
    \label{fig:figure1}
\end{figure}
```

which will produce the following:

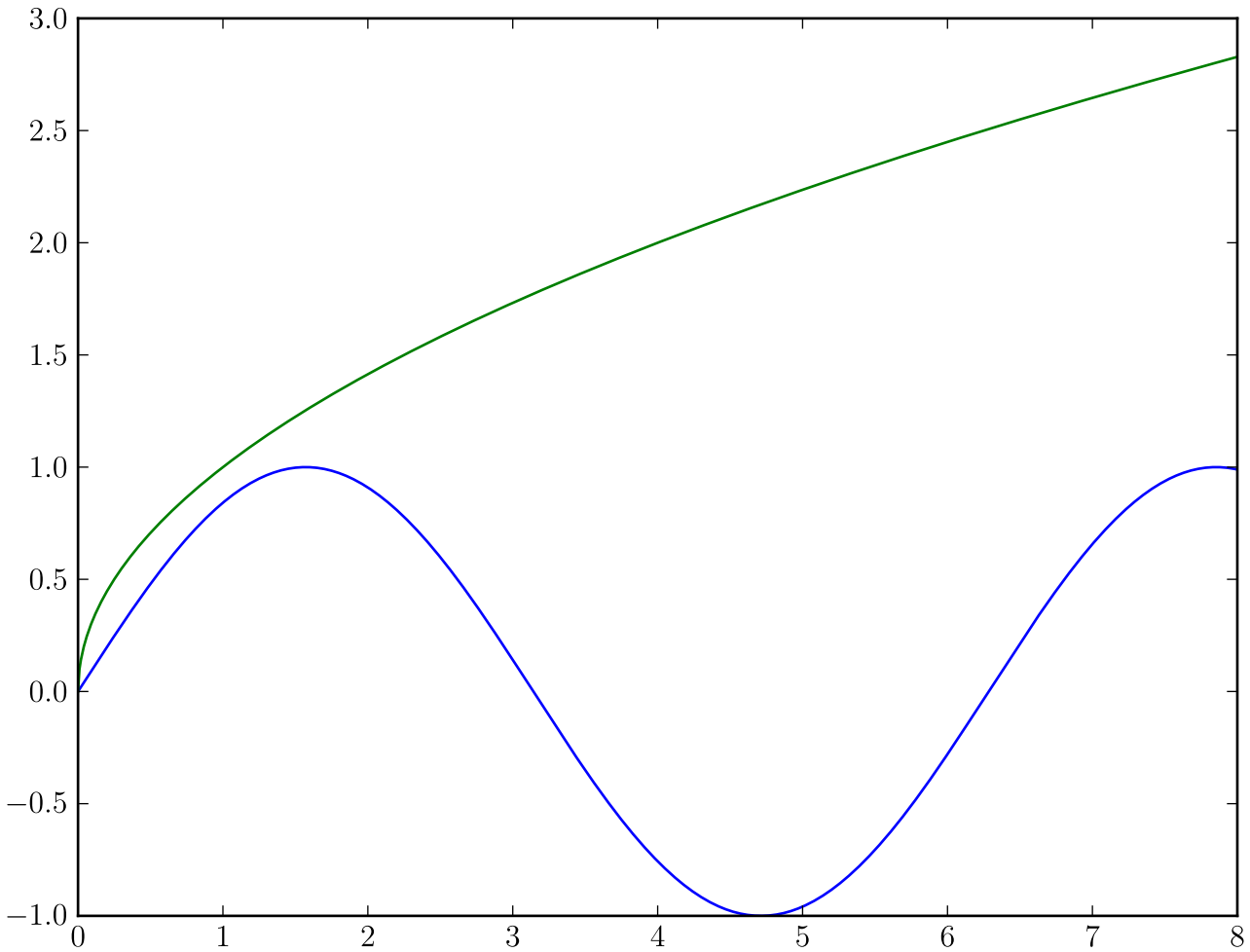

Figure 1: First plot

However one can immediately notice, that the figure is not scaled and the caption text is slightly to far from the actual figure, so these small changes will fix these nuances:

```
\begin{figure}[H]
    \vspace{-5mm}
    \begin{center}
        \includegraphics[width=\textwidth]{plot3.eps}
    \end{center}
    \vspace{-15mm}
    \caption{The same plot as in figure \ref{fig:figure1}, just
       aligned properly.}
    \label{fig:figure2}
\end{figure}
```

And our figure will look as follows:

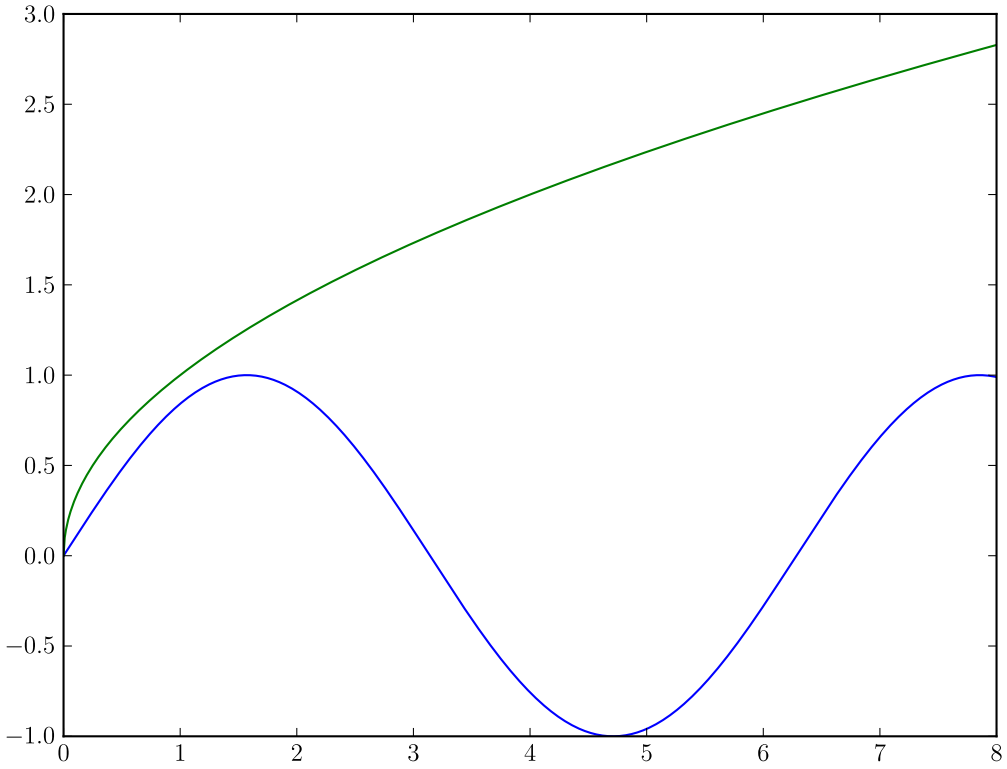

Figure 2: The same plot as in figure 1, just aligned properly.

3 Directory setting for graphics, their formats and LaTeX compilers

The most convenient way nowadays to produce documents from LaTeX source code is to use the `pdftex` or `pdflatex` compiler. It is very convenient as one does not need to convert the `.ps` file every time before viewing it with a viewer. Also there are other benefits to use this compiler.

Using `graphicx` package with this compiler lets you use `.jpg`, `.png` and `.pdf` graphics inside your document. You might ask: "What about my `.eps` figures?" The simple answer is to use `epstopdf` package, which would convert `.eps` figures to `.pdf` figures on-the-fly.

What is more, which `graphicx` package it is possible to give directory names where LaTeX should search for figures. The best example would be the preamble of this file:

```
1 \usepackage[pdftex]{graphicx}
2 \graphicspath{{./figs/}}
3 \usepackage[update,verbose=false]{epstopdf}
```

This means, that the `epstopdf` package will convert the `.eps` graphics only when they are newer than the `.pdf` versions, which might speed up the compiling time significantly.

4 Overlaying a figure with LaTeX code

This is the most interesting capability of LaTeX although it might be considered as 'fiddly' by some users, as it requires some trial and error, but it can be very useful in some cases (eg. Having a lot of figures with chemical structures and getting them cross-referenced and numbered correctly).

The example in the figure 3 just shows how one could do that. It is using the native LaTeX environment `picture` and the only package you need to use additionally is `calc`, so this has very low requirements and is very straightforward.

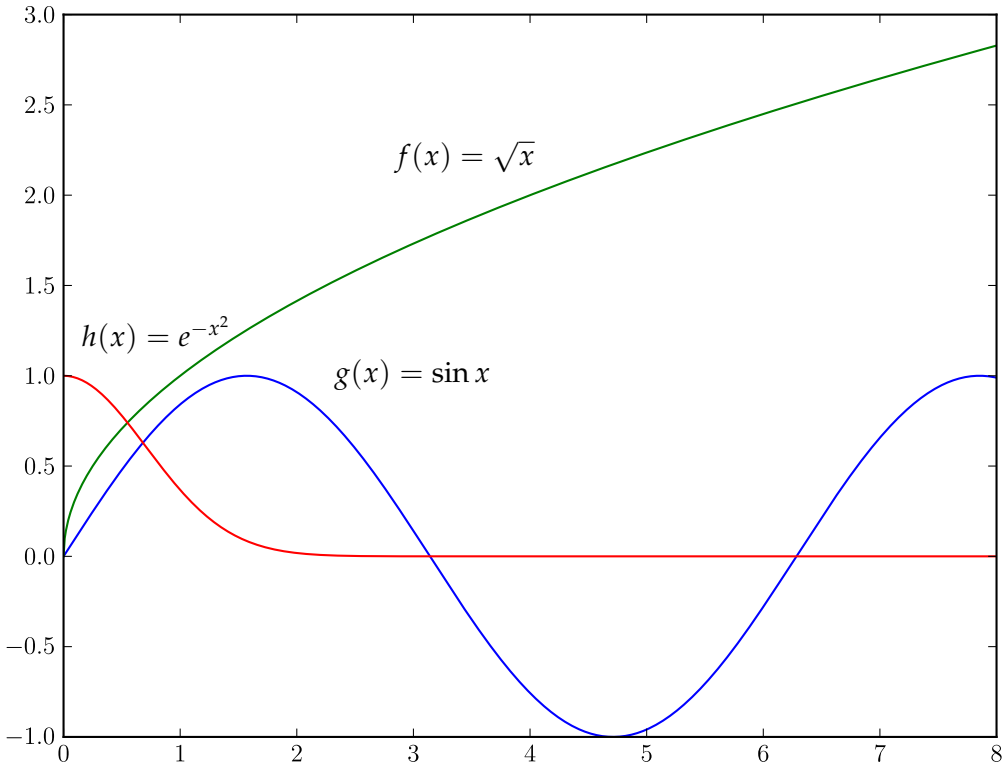

Figure 3: Overlaying LaTeX commands on top of the figure.

```
\begin{figure}[H]
    \centering
    \setlength{\unitlength}{\textwidth}
    \vspace{-5mm}
    \begin{picture}(1,.75)
        \put(0,0){\includegraphics[width=\unitlength]{plot4.eps}}%
        \put(0.4,0.55){$f(x) = \sqrt{x}$}
        \put(0.35,0.37){$g(x) = \sin{x}$}
        \put(0.14,0.40){$h(x) = e^{-x^2}$}
```

```
    \end{picture}%
    \vspace{-10mm}
    \caption{Overlaying \LaTeX\ commands on top of the figure.}
    \label{fig:plot3}
\end{figure}
```

However, there are some nuances. To begin with, you need to know the aspect ratio of your picture, which is not a problem most of the times, since you set the size of your picture your self. If the picture is raster graphics, then the ratio is very easily found by just dividing width and height of the image.