

Getting started with plotting in L^AT_EX

Jason Holloway

This monograph presents a few options for generating high quality figures to accompany scientific publications. The focus of this document is to provide a simple starting point for users with little to no experience plotting outside of the MATLAB environment. The basics of converting a MATLAB figure to tikz are presented. In addition introductory instructions for using tikz and Adobe Illustrator are provided. The techniques herein are only a subset of the tools available to produce high-quality vector graphics. A fully worked example plot and the accompanying code to generate the plot are provided in the GitHub repository. Additional sample plots are provided to illustrate more complex functionality of TikZ and PGF.

I. GETTING STARTED

In order to ensure that I am able to generate figures that are exactly what I intend for I use a number of tools. Not every figure requires using every tool discussed in this paper, simple plots are often ready to go after the first stage.

To follow every step in the tutorial you will need to have the following available on your computer:

- MATLAB
- MATLAB community function matlab2tikz [[Github link](#)]
- MATLAB community function export_fig [[Github link](#)] (optional)
- Working L^AT_EX compiler
- L^AT_EX packages standalone, pgfplots, amsmath, pgfplots. If they are missing the compiler may download them on the fly when they are requested.
- Adobe Illustrator or Inkscape (this tutorial only covers using Illustrator but the same principles apply in Inkscape)

It is assumed that the reader is familiar with plotting in MATLAB and writing and compiling scientific papers in L^AT_EX. Familiarity with TikZ and PGF¹ is not necessary to follow the tutorial; however, creating more advanced plots and tweaking layout, formatting, etc. will require consulting the [online user manual](#). Similarly, experience with Illustrator/Inkscape is not strictly required to follow the tutorial but learning to navigate in the program will help. The tutorial can serve as a guide to help users learn to operate within the program.

NOTE: This document is split into two parts, the first section is a written overview of generating figures while the latter section provides a step-by-step tutorial of creating a figure for publication.

¹henceforth referred to as TikZ

II. GENERATING PUBLICATION QUALITY PLOTS

A. Disclaimer

Everything is still a work in progress as I learn better techniques and become more proficient with each of the tools, a current sampling of my workflow follows. Please note that I only do all of these steps for plots that are ready for publication. Temporary plots are still just saved from MATLAB and tossed into the paper.

Keep in mind:

- Both TikZ and Illustrator/Inkscape have steep learning curves to master. However doing simple modifications is not too hard.
- The first time you run through this process will take a while. There are a lot of elements that need to be configured for the first time. I would recommend trying this on some dummy data before attempting to create a plot for publication during a deadline rush.
- Creating good plots can take a long time. It is most important to plan what a figure will look like before going through this pain.
- There may be better alternatives available. I have arrived at this hodge-podge of steps through a meandering procession from MATLAB to Excel to TikZ to Illustrator to online tools. Each has their merits.
- A benefit of finally having a vector graphics file in PDF is that it is fairly trivial to go back and tweak the appearance. It is worth the time to get everything set up so that last minute changes are easily accommodated.
- Another benefit of using Illustrator/Inkscape is that it is easy to composite a complex figure that has multiple parts if everything is already a PDF. (I think) Illustrator has the ability to plot data directly from within the program. I have yet to use this feature; perhaps it is worth a look.

On to my very simple² three part process.

B. MATLAB

Plot the data in Matlab and make any broad changes to the plot that I want to create. This generally means fixing the axes limits, trying different line thicknesses for the plots, and playing with line styles (dashed, dotted, etc.).

There are two important parameters at this stage, the axes limits and the general aspect ratio of the plot. They can both be fixed and tweaked in the next step, but everything is a little easier if these are in the correct ballpark.

C. TikZ

I use a tool such as matlab2tikz to get the data out of MATLAB and into a TikZ format. The tool can be a bit of

²relative to something

overkill with all of the settings that it preserves. I find that it is faster for me to use `matlab2tikz` and delete the extra stuff than it is for me to try to build up a full plot from scratch.

It is at this step where you face the biggest choice. Either spend time now to get the composition correct or spend time in Illustrator or Inkscape to get the composition correct. I currently find that it is more beneficial to use `TikZ/pgfplots` to get the plots to be nearly perfect and to use Illustrator to move labels, tweak colors and thicknesses, etc.

- 1) In MATLAB: Make sure the plot you want is in focus then switch to the command line. Call `matlab2tikz` to export the figure. An example call that I use is `matlab2tikz('test.tikz', 'standalone', true, 'externalData', true)`. This creates a file called `test.tikz` and N data files (`test-1.tsv, ..., test-n.tsv`), one data file for each data series in the plot. Using external data files keeps `test.tikz` clean.
- 2) In \LaTeX : Open `test.tikz` (you can name it `test.tex` if you prefer) in your \LaTeX editor of choice. Compile the file to make sure everything is working and to get an idea of how the plots are going to look.
- 3) In `test.tikz`, be sure to set the dimensions of the plot to match the dimensions of the figure when it is placed in the paper. For example, a two-column paper will often have figure widths of ~ 3.5 in and ~ 7.15 in (including the gutter between the columns). This is an important step so that you know the text sizes are correct.
NOTE: The 'height' and 'width' options in `TikZ` refers to the size of the plot area only. The full figure size will be larger so compensate accordingly.
- 4) Remove all of the extraneous stuff from the `TikZ` file that you don't want. `matlab2tikz` includes a bunch of settings that MATLAB has enabled (e.g. the tick mark colors in MATLAB are included during export).
- 5) Make any remaining tweaks to the plots that you find necessary. It is possible to move text labels closer to/further from the axes. Similarly, the legend (if included) can be moved around. Tick marks can be manually set if desired (both positions and labels) though the automatic numbering is usually pretty decent.

For simple plots converting to `TikZ` may be sufficient and you can stop here. If everything looks good and seems to fit in the designated space then you may not need to tweak further in Illustrator.

D. ILLUSTRATOR/INKSCAPE

For complex plots, or for plots with legends/labels that I cannot get to format correctly in \LaTeX I often use Illustrator or Inkscape to polish the figure. More complicated figures with multiple parts (A, B, C, etc.) are almost certainly easier to compose using Illustrator than trying to get everything to work properly in \LaTeX or `TikZ`. I find Illustrator to be a better tool than Inkscape, but Inkscape is free and doesn't require a virtual machine to run on Ubuntu. Use whichever you find most comfortable.

In order to edit the PDF generated by \LaTeX you need to install the fonts used by \LaTeX on your system. It's a bit of a pain the first time, but a combination of Google, Stack Overflow, and persistence will help win the day. On a Windows OS the fonts are (roughly) located in `C:\Program Files\MiKTeX 2.9\fonts\type1\public\amsfonts\cm\`. You don't need to install all of the fonts, but that can be the easiest thing to do. You will know if a font is missing when you try to open the PDF in Illustrator/Inkscape and the program gives an error about missing fonts. (An example of this is included in the tutorial.)

Once in Illustrator ³, you may find that you cannot easily change elements of the file. This is because \LaTeX includes a ton of clipping masks when it creates a PDF. So either, keep clicking on the element you want to move (effectively entering a new sublevel with each double click) OR remove all of the clipping masks ([old example here](#)).

From here you can use the select tool (v) to highlight and move elements around. You can group elements to move together, align elements etc. I find that it is easy to use Illustrator to modify plot colors, dash spacing, and line width.

NOTE: It is extremely important to save a copy of the PDF you intend to modify (or change the file name) so that if you rerun \LaTeX (to add an additional plot for example) it does not override your changes in Illustrator. Only open the PDF that does not share the filename of your `TikZ` file. In this example, do not open `test.pdf` in Illustrator, make a copy `test_ai.pdf` and open that instead.

III. EXAMPLE FROM MATLAB TO TIKZ TO ILLUSTRATOR TODO

Examples: I have attached my tutorial and a few examples. Some of these examples are more complex than others. I suggest starting small and only using `TikZ` to generate plots. I wrote my first journal paper only using `TikZ/pgfplots` before I even knew how to use Illustrator.

In the tutorial, I work through an example of a representative plot. Included is the standard figure produced in MATLAB, the original `TikZ` file from `matlab2tikz`, the cleaned `TikZ` file, and finally the end product after Illustrator. I also include examples of plots I have created for my friends to transform their MATLAB output into vector graphics.

³I will assume Illustrator is used from now on, but the concepts are similar in Inkscape.