

*L^AT_EX*Lecture 11

Justin A. James

Minnesota State University Moorhead

<http://web.mnstate.edu/jamesju/Spr2017/M291-01S17.html>

jamesju@mnstate.edu

April 10, 2017

1 *TikZ*

2 *Example 1: Some Basic Shapes*

3 *Example 2 Graphs*

4 *Using Other Programs*

- 1 *TikZ*
- 2 *Example 1: Some Basic Shapes*
- 3 *Example 2 Graphs*
- 4 *Using Other Programs*

What is TikZ?

TikZ stands for “TikZ ist kein Zeichenprogramm”, which translates to “TikZ is not a drawing program”.

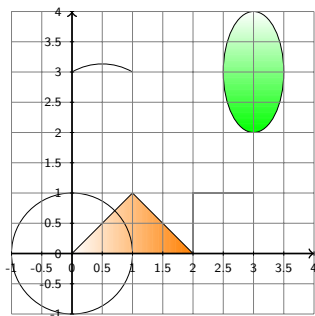
But TikZ IS a drawing program with a lot of bells and whistles.

This lecture is based on some of the examples from the TikZ manual. Here is a link to a nice brief introduction to TikZ.



- 1 *TikZ*
- 2 *Example 1: Some Basic Shapes*
- 3 *Example 2 Graphs*
- 4 *Using Other Programs*

Example 1 Result



The TikZ Environment

Start a new document. In the preamble type

```
\usepackage{tikz}
```

```
\usetikzlibrary{calc,intersections,through,backgrounds}
```

When you want to create a TikZ picture use the following commands:

```
\begin{tikzpicture}
```

```
\end{tikzpicture}
```

Add these commands to your document in the body.

Drawing Basic Objects

One of the basic commands in TikZ is

```
\draw
```

Add the following text to your tikzpicture:

```
\draw (0,0) -- (1,1) -- (2,0);
```

Notice that at the end of a command (some commands can be quite long) there is always a semicolon.

Feel free to add more points and see what happens.

Also, observe that you don't have to tell tikZ right away how much space you'll need. It just makes the space it needs on it's own.

Drawing Basic Objects

Suppose we wanted to complete our shape to a triangle. We use the command:

```
\draw (0,0) -- (1,1) -- (2,0) -- cycle;
```

In the picture at the beginning, the triangle was shaded orange. To shade an object we use the `filldraw` command.

Copy your `draw` command and paste it so that you have

```
\draw (0,0) -- (1,1) -- (2,0) -- cycle;
\filldraw[orange] (0,0) -- (1,1) -- (2,0) -- cycle;
```

You should have an orange triangle with a black outline. It's not necessary to use two commands to get the black outline.

Change your `filldraw` command to

```
\filldraw[fill=orange, draw=black]
(0,0) -- (1,1) -- (2,0) -- cycle;
```

Drawing Basic Objects

Some other commands for drawing are.

- draw (h,k) circle (r) ; where (h,k) is the center and r is the radius
- draw (h,k) ellipse $(m$ and $M)$; where (h,k) is the center, m (M) is half the length of the axis in the x (y) direction.
- draw (x_0,y_0) arc $(a:b:r)$; where (x_0,y_0) is the starting point, a is the starting angle, b is the ending angle and r is the radius.
- draw (s,w) rectangle (n,e) ; where (s,w) is the southwest corner of the rectangle and (n,e) is the northeast corner of the rectangle.

Basic Objects Practice

Add these commands in your document.

```
\draw (0,0) circle (1);  
\draw (1,3) arc (60:120:1);  
\draw (2,0) rectangle (3,1);  
\draw (3,3) ellipse (0.5 and 1);
```

You can specify the units used to measure the radius: cm, pt, in, ...

Grid Construction

You can quickly and easily create a coordinate grid in your picture with the following command.

```
\draw[options] (s,w) grid (n,e);
```

Try the following.

```
\draw[step=.5] (0,0) grid (4,4);
```

The lines on the grid are so dark and thick that they are distracting. Try the following.

```
\draw[step=.5,gray,very thin] (0,0) grid (4,4);
```

Arrow

It might be nice to have some axes on our grid, which means we need some arrows.

Add the following commands.

```
\draw[->,thick] (0,0) -- (4,0);  
\draw[->,thick] (0,0) -- (0,4);
```

Nodes: Adding Text

In order to add text we need the next basic command in TikZ: node. TikZ is especially useful for drawing graphs and diagrams. (It's not very good at graphing functions.) Nodes and edges are the fundamental building blocks of a graph.

Add the following command to your tikzpicture.

```
\draw (0.5,1pt) -- (0.5,-1pt)
node[anchor=north] {\small 0.5};
```

We could continue the process of labeling by adding the command:

```
\draw (1,1pt) -- (1,-1pt) node[anchor=north] {\small 1};
```

Adding all labels this way seems like a lot of work. Tikz makes this easier.

For Loops

The syntax for for loops in TikZ is as follows.

```
\foreach \x in {values x can take on}
whatever you want to draw;
```

For example, to draw the rest of our axis labels we would add the command

```
\foreach \x in {0,0.5,...,4} \draw (\x,1pt) --
(\x,-1pt) node[anchor=north]{\small \x};
```

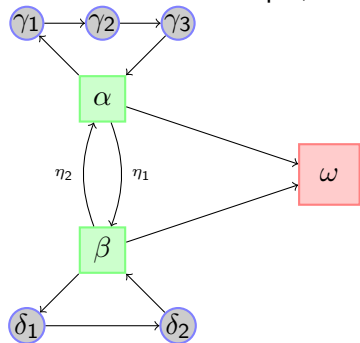
Try labeling the y-axis. (Hint: you will want to anchor your nodes in a different direction).



- 1 *TikZ*
- 2 *Example 1: Some Basic Shapes*
- 3 *Example 2 Graphs*
- 4 *Using Other Programs*

Introduction to Graphs

For this second example, our goal is to create the following graph.



Ode to Nodes!

There are many different ways to designate nodes. We are going to learn one of them and build on it.

The command

```
\node at (x,y) {};
```

will put an empty node at the coordinates (x,y).

Create a new tikZ picture on a new page and add the commands

```
\node at (0,0) {};
```

Let's make this a friendly node by typing Hi inside of the {}.

Jazzing Up Our Nodes!

We want to put some kind of shape around our nodes. We can put either a circle or a rectangle around the node. The command is

```
\node at (x,y) [rectangle,draw=color] {};
```

```
\node at (x,y) [circle,draw=color] {};
```

Change your command to have a black rectangle around the node.

To add a background color add the optional command `fill=color` after the draw command.

Change your command to have a green background.

Jazz Hands!

We can also blend colors. Suppose we wanted a darker green for a background than we go. We could use the command `fill=green!50!black`.

So it's a bit dark. Change it to `green!70!black`.

The default second color in a blend is white, so typing `fill=green!50` will produce something that is 50% green and 50% white.

If we want to have a color gradient, we can use the `\shadedraw` command. Our very first drawing used the command

```
\shadedraw[left color=white, right color=green, draw=black]
```

to create the color gradient in the triangle. A similar command with a bottom color and a top color was used to fill the ellipse.

THESE are Spirit Fingers!

Let's add a couple more nodes to our picture.

Using the following locations just add the nodes in the picture.

Note that math mode is **not** assumed within the node label command.

- ω is at (3,0)
- α is at (0,1)
- β is at (0,-1)
- γ_1 is at (-1,2)
- γ_2 is at (0,2)
- γ_3 is at (1,2)
- δ_1 is at (-1,-2)
- δ_2 is at (1,-2)

Style Baby!

We want certain nodes to look the same, so we will create a style for those kinds of nodes. The command for style looks like this

```
\begin{tikzpicture}  
[nameofstyle/.style={what your style looks like}]  
\end{tikzpicture}
```

What? One does want a hint of Color.

Let's create the style for the α and β nodes. Add the command right after the begin tikzpicture command.

```
[source/.style={rectangle,draw=green!50,fill=green!20,thick,
inner sep=0pt,minimum size=6mm}]
```

Most of this we know, but some is new.

```
minimum size=6mm
```

creates a minimum dimension of the rectangle. This gives a uniformity to the dimensions of the nodes with this style.

```
inner sep=0pt
```

makes sure that if your minimum size is really small there will still be space for added text.

That Lumpy Blue Sweater

Now, change your α and β nodes to

```
\node at (0,1) [source] {$\alpha$}
\node at (0,-1) [source] {$\beta$}
```

The other two styles will be called sink and build. Add these to the optional commands at the beginning of your picture.

```
[source/.style={rectangle,draw=green!50,fill=green!20,thick,
inner sep=0pt,minimum size=6mm},
build/.style={circle,draw=blue!50,fill=black!20,thick,
inner sep=0pt,minimum size=4mm},
sink/.style={rectangle,draw=red!50,fill=red!20,thick,
inner sep=0pt,minimum size=8mm}]
```

Now, add these styles to the appropriate nodes.

What is your Quest?

We can also name nodes too! Let's give our nodes some descriptive names. For example we'll change the commands for α and β to

```
\node (alpha) at (0,1) [source] {$\alpha$}
```

```
\node (beta) at (0,-1) [source] {$\beta$}
```

Go through and name the rest of your nodes in a similar way

Toe Pick!

Names are useful because they give us a way to draw edges. As with node commands there are many different ways to create edges. We are going to focus on one of them. Change the command for `gamma1` to

```
\node (gamma1) at (-1,2) [build] {$\gamma_1$}
edge[<-] (alpha);
```

This command draws an edge from α to γ_1 .
Try adding all but the curved edges.

I'm a donkey on the edge!

To draw the curved edges add the following command to the beta node line

```
edge[<- ,bend right=20] (alpha) edge[-> ,bend left=20] (alpha)
```

The bend command bends the arc in the direction you want it to bend by the degrees you specify.

Edge of Tomorrow

To add text to an edge, we create a node by the edge. Change your curved edge commands to

```
edge[<-,bend right=20] node[auto,swap] {\tiny $\eta_1$}
```

(alpha)

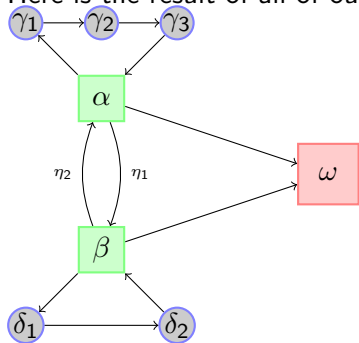
```
edge[->,bend left=20] node[auto] {\tiny $\eta_2$}
```

(alpha)

The auto command tells tikZ not to put the node right on top of the edge. The swap command changes the side of the edge where the node is placed.

The Result

Here is the result of all of our hard work.





- 1 *TikZ*
- 2 *Example 1: Some Basic Shapes*
- 3 *Example 2 Graphs*
- 4 *Using Other Programs*

Other Programs

While TikZ is fun to use, some people don't like it because you still have to create figures using code. This is also true of pstricks. However, there are programs out there where you can “draw” pictures, save them and then import them into \LaTeX .

- IPE (Windows and Mac versions exist, Free, Good for basic geometric shapes and graphs)
- Xfig (similar to IPE, Unix Based, Free)
- Winfig (Rip off of Xfig, Windows Based, Not Free)
- Maple (Good for graphing functions, not free (very expensive))
- Mathematica (Similar to Maple)
- SAGE (Free, Open Source, similar to Maple and Mathematica but a bit harder to use.)

What is IPE?

IPE stands for Integrated Picture Environment. It is a WYSIWYG (What You See is What You Get) drawing program that interacts well with \LaTeX .

To download IPE, use this link and scroll down to find the appropriate binary file (Linux, Windows, or Mac OS).

Extract the files in the compressed folder to a convenient directory. The executable file for IPE version 7.2.7 can be found inside the bin folder.

You can find the IPE user manual here.

There are many drawing programs that can be used to create graphics for inclusion in a \LaTeX document. For this week's lab assignment, I will give extra credit to those who find and use an external program to create and include an image – you must create it yourself. Please include information the program used to create your image and, if possible, the original file from the program you used.