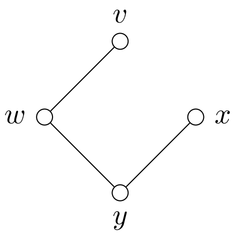
Drawing Graphs using TikZ in LaTeX

Max Hailperin, Gustavus Adolphus College

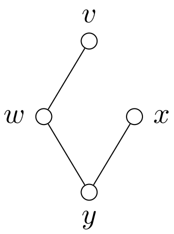
MCS-236, Fall 2011

This document illustrates some particular techniques I've found convenient for drawing graphs. They aren't by any means the only possibilities, or even necessarily the best. All the example graphs come from our textbook, Chartrand and Zhang's *Introduction to Graph Theory.* You should read the LaTeX source file for this document in parallel with the resulting PDF file, so that you can see what code was used to produce each drawing.

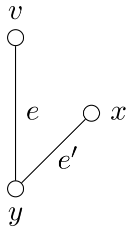
Let's start with the graph H from Figure 1.15 on page 10. Notice that drawing has the vertices aranged in regularly spaced columns and rows. If we count columns from the left, we can see w as in column 0, v and y as in column 1, and x as in column 2. Likewise, counting rows up from the bottom, y is in row 0, w and x are in row 1, and v is in row 2. Using these as the coordinates in a rectangular (Cartesian) coordinate system, we can draw the graph as follows:



Comparing this result to what is in the book, the big difference is that the one in the book seems squished horizontally. We could fiddle with the coordinates to achieve this effect, but a better approach is to leave them as the nice simple integers 0, 1, and 2 and instead change the unit of measure for the coordinate system. By default, each unit in both dimensions is 1cm. But we can change that, using a smaller unit in the horizontal direction. Following convention, the horizontal and vertical directions are refered to as the x and y dimensions, not to be confused with the vertices that happen to also be labeled x and y in this particular graph. Here's the version with a reduced x unit:

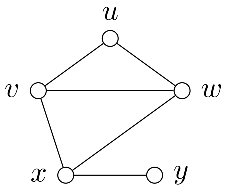


Continuing with another example from the same figure, the graph F' lets us see how to put labels on edges. Note that the `pos=.4,` part can be left off, and the drawing still is OK, it just doesn't look quite as right to my eye. (Try it.) I initially did the drawing without it, and then when the e' looked a bit mispositioned, added that extra positioning fudge in to make it look right. The number indicates what fraction of the way along the edge the label appears; the default is .5, that is, midway. Another item to notice in the code is that I copied as much of it as possible from the prior example. As such, there are no vertices with x-coordinate 0. That's OK; you can use whatever range of coordinates is convenient, and the drawing will still be centered.

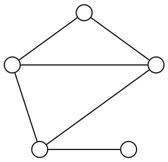


For highly symmetric drawings, it is often easier to use polar coordinates. For example, Figure 1.3(a) on page 3 has the following graph with five vertices arranged evenly spaced, as though they were each 1/5 of the way around a circle. Because a circle has 360 degrees, this means the angle to each point needs to be $360/5 = 72$ degrees further rotated from the previous one. For example, we could get even spacing with angles of $0, 72, 144, \ldots$ degrees. But if we did that, the whole drawing would be rotated a bit compared with the one in the book. (You can try it.) In particular, it wouldn't have the u vertex straight up at the top of drawing, because it would be at a 72 angle from the positive x-axis instead of 90 degrees. If we want 90 degrees, then the preceding vertex, w, better be at $90 - 72 = 18$ degrees. Thus our progression of angles in degrees is $18, 90, 162, 234, 306$.

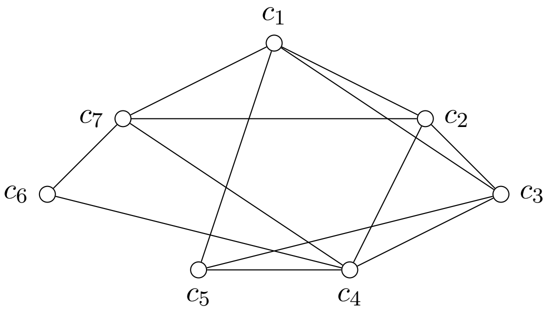
Putting each of these together with a radius of 1 results in the following graph:



Part (b) of the same figure shows an unlabeled version of the same graph, which can be made simply by leaving out the portion of the code that does the labeling:



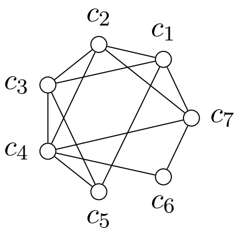
The very first graph shown in our textbook, in Figure 1.1 on page 1, will serve as an example of how a drawing might be improved. Initially we can try to reproduce the drawing more or less as shown in the book, using rectangular coordinates as in the first example:



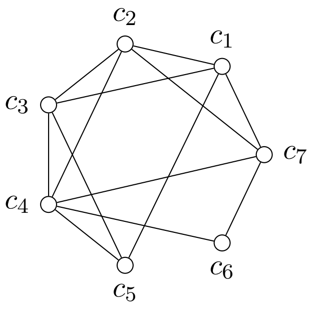
As a brief aside, note that in the source code for the labels, I used braces around the subscripts, such as the 6 in c_6. That isn't actually necessary so long as each subscript is only a single digit. But it's a good habit to get into, so that in larger cases you get c_{12} instead of c_12.

Sometimes explicit positioning of vertices onto a rectangular coordinate system lets you find an arrangement where the structure of the graph is clear. (Consider, for example, the bipartite graphs in Figure 1.26 on page 21. The bipartite structure is much clearer on the right of that figure, where the vertices are arranged into two rows.) But in the particular example of

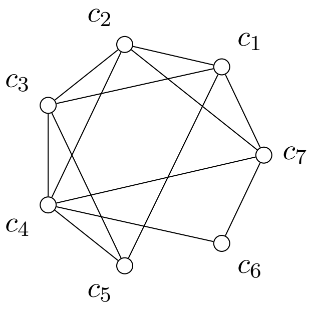
Figure 1.1, I don't see any obvious structural regularity popping out, so the drawing doesn't seem to gain anything from the manual layout. In cases like that, my first instinct is to try positioning the vertices equally around a circle, as with the earlier five-vertex example. Here we've got 7 vertices, so each one would be something like 51.4 degrees from the previous one. For the human eye, we can probably round the angles off to integer numbers of degrees, alternating between spacings of 51 and 52 degrees:



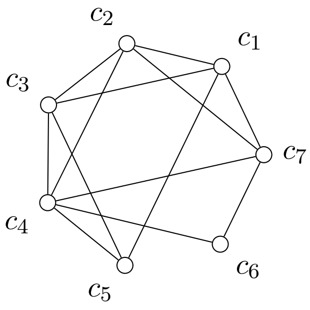
In a bit we'll see a way to make the vertices less tedious to specify, but so far as the resulting drawing goes, the only real problem with the previous attempt is that it is rather tightly scrunched together. That could be fixed by changing the radius on each point from 1 to 1.5. However, as in our first example, it makes more sense just to change the coordinate system, in this case stretching both the x and y dimensions equally:



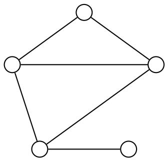
One of the tedious aspects of the preceding drawing was figuring out for each vertex whether its label should appear above the vertex, to the left, below, or to the right. In many cases there is no one unique answer to that question. For example, moving c_1's label from above the vertex to instead be at its right would be entirely plausible. On the other hand, it is easy to get one of the positions indisputably wrong, for example by positioning c_1's label below the vertex or to its left. In drawings with the vertices arranged around a circle, such as this one, one easy alternative is to position each label in accordance with the vertex's angle:



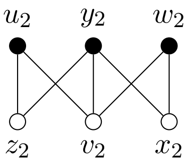
Opinions will differ as to whether that version is more or less attractive or clear than the preceding one, but it certainly requires less thought to produce. Finally, we can notice that all seven vertices fall into a common pattern. Rather than repeating the same code seven times over, we can use a loop to produce them; while we're at it, we'll let the computer do the arithmetic to calculate the angles:



The same approach could be used for the earlier five-vertex graph. This relies upon the fact that the `foreach` loop can loop over names as well as numbers. However, we still need the numbers in order to calculate the angles. Luckily, `foreach` can step a pair of variables in tandem, letting us produce the following example. (You can add the labels yourself if you want them.)



Returning to rectangular coordinates, I earlier mentioned the bipartite graphs in Figure 1.26 on page 21 as good examples where laying out the vertices in rows was helpful. If you look at the drawings on the right of that figure, you'll notice that they contain one other visual element, which is that some of the vertices are filled in. Here, this serves as way of distinguishing one subset of vertices from the other. Elsewhere, the same visual style is used for other reasons. For example, in Figure 4.5 on page 89, the filled vertex is the root of the tree. Here is one of the drawings from Figure 1.26 illustrating how some vertices are filled:



All of the examples up until now have been ordinary, undirected graphs. For most of the course, that's all we'll work with. But there will be approximately a week in which we work with digraphs, that is, directed graphs. In the textbook's drawings of digraphs, each directed edge has an arrowhead somewhere in the middle of it. Another very common style of drawing, where the arrowhead is at the end of the directed edge, is easier in TikZ. However, the book's mid-edge style can also be achieved, as shown here. This example, based on the digraph D_1 in Figure 1.37 on page 28, also illustrates one way of bending an edge into a curve. Sometimes that's helpful even in undirected graphs, in order to improve visual clarity.