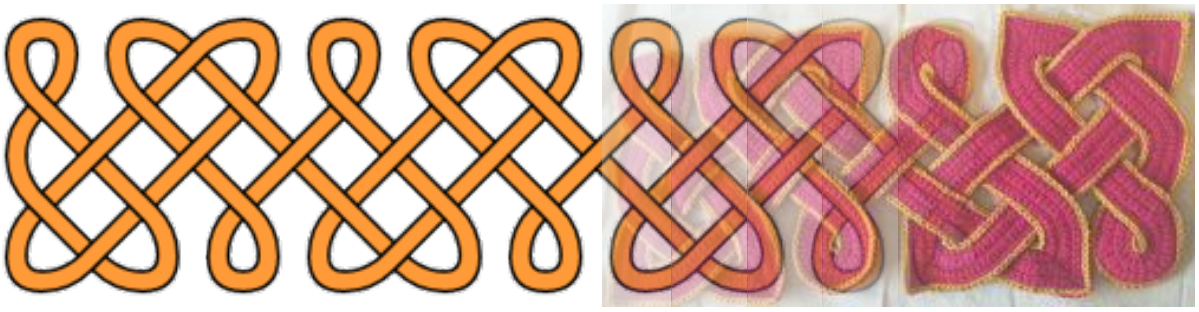
A General Method for Crocheted Knotwork

Elyse Yeager

This document typeset June 23, 2021.

Contents



Chapter 1

Introduction

Many knot motifs are drawn in a rectangular shape, with strand crossings at regular intervals. Since these knots are arranged from a small collection of components (straight lines, quarter circles, etc.), if you know how to crochet each component, you can crochet a wide variety of knot designs.

This document gives explicit instructions for crocheting two example knots in Chapter 2. A method for drawing knot motifs is in Chapter 3. Then there are two levels of explanation detailing how to turn knot motifs into crochet patterns. The first, Chapter 4, gives instructions for identifying the various components that make up a knot. In that section you will find a link to some basic Python code to automate the most tedious parts of turning a list of components into a pattern. (If you'd rather do that step on your own, Appendix A has instructions for making each component.) Chapter 5 is where I "show my work," explaining the geometry I chose for each component. Understanding the content of Chapter 5 is not necessary for generating crochet patterns.

Chapter 2

Instructions for Making the Example Knots

Section 2.7 has a phototutorial for the rectangle example knot, illustrating the major steps described below.

2.1 Materials

In addition to yarn and hooks, you'll want safety pins, a tapestry needle, lengths of contrasting yarn for running stitch markers, and masking tape.

2.2 Overview

Each strand of the knot pattern is made of rows of single crochets, with strategic increases and decreases to cause the strand to curve as desired.

Figure 2.1: Strand closeup with running stitch markers

Instructions are given for crocheting the strands. There are usually several strands per knot. These are worked separately, then woven together.

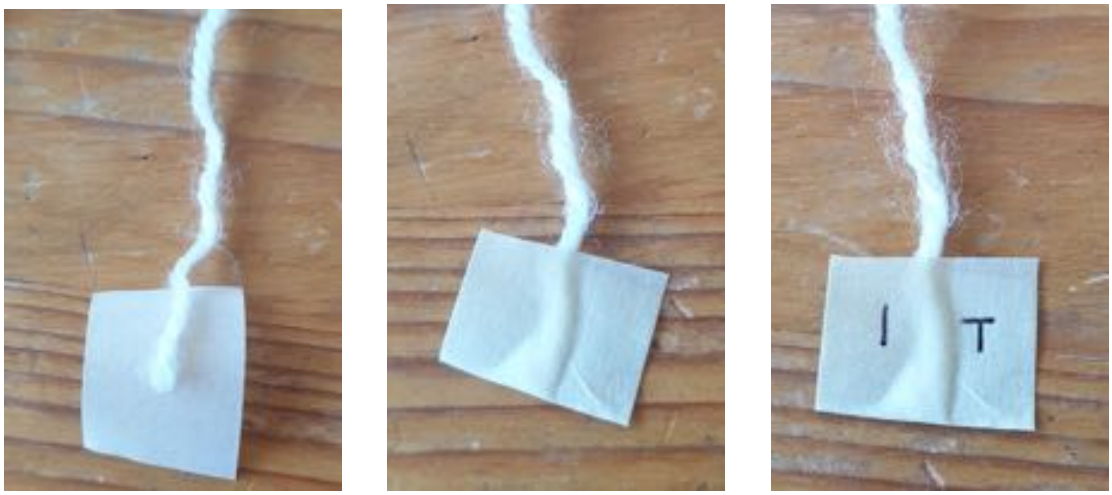
To help with assembly, the intersections (where a strand crosses itself or another strand) are labelled. Each pattern has a picture of the finished knot with labelled intersections. These labels match the marker

labels given in the foundation row instructions. "T" means the strand is on top and "B" means it is on the bottom. Once the assembly is complete, these markers are removed.

To keep the knot aligned, it is helpful to secure the crossings. So, the last step in assembly is to sew along each strand.

2.3 Running Stitch markers

To make labelled markers, fold a piece of masking tape over the end of a scrap of yarn, and write the label on the tape. Be careful to test whether your pen[1] will smudge before you put the markers near your work.



(a) Stick end of yarn to tape (b) Fold tape over (c) Label

Figure 2.2: Making labelled markers

It's nice to make all the markers for a strand (both labelled and unlabelled) before you start the foundation row, and lay them out in order.

(a) Stick yarn scraps to a length of masking tape at 1 cm intervals

(b) Fold the tape over and press the sides together well

(c) Write labels and cut the markers apart

Figure 2.3: Making many labelled markers at once

[1] I use Frixion pens, whose ink disappears with heat. First, I don't have to worry about stray marks on the yarn. Second, I can re-use the markers by heating them up on an electric heating pad. If you only make one knot, re-using labelled markers isn't practical, but if you plan on making more, it's nice to save the time and materials.

Labelled stitch markers show you the exact locations were the strands cross, so they need to be left in the work until it is assembled. The unlabelled makers are only needed while the strands are being made, so they may be removed and reused in different strands.

Markers are added to the piece when the foundation chain is being made.

(a) Step 1: work the required number of chain stitches before the marker

(b) Step 2: lay the marker over the working yarn

(c) Step 3: continue making chains, trapping the marker in place

Figure 2.4: Placing markers in foundation row

In subsequent rows, when you come to a marker, flip it from the back of the work to the front. This will

result in a dashed line running from the first row to the last.

(a) Step 1: work the required number of stitches before the marker

(b) Step 2: flip the marker from back to front, just after the last sc you worked

(c) Step 3: continue work, trapping the marker between two stitches

Figure 2.5: Running stitch markers in subsequent rows

2.4 How to Read the Patterns

Segments Each strand of the knot is broken into segments, separated by running stitch markers.

Running stitch markers In foundation rows, unlabelled stitch markers are shown with a plain slash: /
while labelled stitch markers are shown with their label, e.g. $\overset{\text{1T}}{/}$. In subsequent rows, all markers are indicated by a plain slash.

Labelled markers need to be kept in place until the knot is assembled. Unlabeled markers may be removed and reused after its strand is crocheted but before the strands are woven together.

For small knots, you can probably omit the labels on the markers and figure out the weaving just fine. For larger knots, though, they come in very handy.

Foundation row The numbers listed in the foundation row refer to chain stitches. Markers with a superscript are labelled, and the superscript gives the label.

For example, if you see this:

3 / 4 $\overset{\text{1B}}{/}$ 5 $\overset{\text{2T}}{/}$ 6

then you should work the following:

`chain 3, place an unlabelled marker, chain 4, place a marker with the label "1B", chain 5, place a marker with the label "2T", chain 6`

Turning Between each row, ch 1 to turn. The turning chain never counts as a stitch and is not written in the line-by-line instructions.

Subsequent rows Unless otherwise noted, the numbers listed refer to single crochet stitches. To make the pattern more readable, the labels of stitch markers are no longer given.

If you see this, and you are not in the foundation row:
`5 / 6 / 1 sc , 2 ch, 1 sc / 7`

it means this:
`5 sc, flip marker, 6 sc, flip marker, 1 sc, 2 ch (do not skip any stitches), 1 sc, flip marker, 7 sc`

Increasing and Decreasing To shape the strand, you will work increases and decreases. Increases are almost always[2] worked as (2 sc in one stitch), and decreases are almost always worked as slst2tog.

Curved Segments The locations of increases and decreases within a segment are usually not given explicitly. Rather, you should space them out roughly evenly. So, if you work one increase on one side, work a subsequent increase on a later row on the other side.

The number of increases and decreases are given in parentheses. For example, if you see this:
`8 (-1)`

it means this:
`The next segment will have 8 stitches total, which requires one decrease. That is, the segment currently has 9 stitches; in this row you will make 7 sc and one slst2tog.`

`If you are in Row 1, the slst2tog can go anywhere. For subsequent rows, look for a portion of the segment that is flatter than the others. Work the slst2tog there to curve it.`

Similarly, if you see this: `8 (+1)`

it means this:
`The next segment will have 8 stitches total, which requires one increase. That is, the segment currently has 7 stitches; in this row you will make 8 sc, two of which will be in the same stitch.`
`If you are in Row 1, the increase can go anywhere. For subsequent rows, look for a portion of the segment that is flatter than the others. Work the increase there to curve it.`

Pointy Segments There are places where the strand forms a right-angled point. These are worked somewhat differently from the curvy bends. For that reason, the locations of increases (in the form of chains) and decreases (in the form of slst2tog's) are explicitly specified in the instructions. So if you read something like "`3 sc, 2 dec sts, 3 sc`" then work them exactly as they are written, rather than trying to sprinkle the decreases around the segment.

After making chain stitches, in subsequent rows, work into the stitches themselves, rather than the chain space. Do not skip a stitch when you make a chain.

Ending After the end of the last row, leave a tail for stitching the two ends of the strand together.

2.5 Assembly and Finishing

2.5.1 Outlines

After you've worked the last row of a strand, turn it and slip stitch in each single crochet in a contrasting colour. Do the same along the foundation row, so there is a slip-stitched outline along each edge of the strand. Leave a tail where you finish each slip-stitch outline.

When you start, be careful to work into the *backs* of the last row of sc. Otherwise your outlines will show up on the wrong side of the knot. Also be careful to work loosely, so the strand does not warp or curl. At the insides / outsides of sharp points, you may want to slst2tog / add a chain between slip stitches.

This outlining serves several purposes. It makes the two sides of each strand more symmetric (instead of one side being obviously a chain, and one side sc). It stiffens up components, such as points, that might

[2] In rare cases where there are more increases than existing stitches, you may have to work (3 sc in one stitch) or slst3tog.

otherwise curl in on themselves. It also highlights each strand – this is more important for thick strands than for thin ones. In a knot with thick strands, the knot pattern can be lost or indistinct without clear borders.

The outline is very difficult to add after the knot has been woven together, so it's best to add it before continuing on to the next step.

2.5.2 Weaving

Each pattern comes with a picture of the knot with numbered intersections. The numbers match up to the numbers on the stitch markers. The markers are additionally labeled T and B, to indicate which goes on top and which on the bottom.

Matching up markers, weave the strands into place. At each intersection, the labelled markers should cross at right angles, meeting in the centre of each marker.

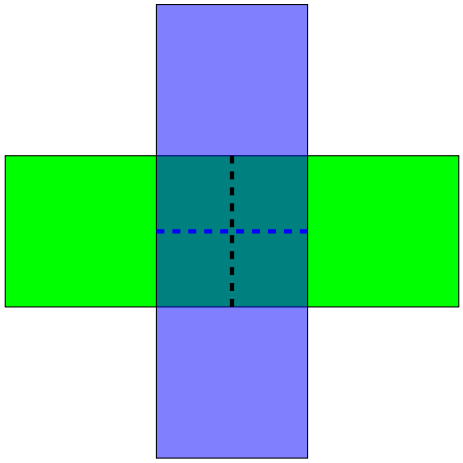


Figure 2.6: Lining up intersections. The green strand has the vertical running stitch marker, and the blue strand has the horizontal running stitch marker

Use a safety pin to attach the top and bottom strands until they can be permanently secured together. The start and end of each strand will be cross underneath another strand, to hide the join. Stitch the two ends together using the tail left over when you broke the yarn. Finally, use the tails of the outlines to connect the two outlined edges.

2.5.3 Sewing

The strands can slip and bunch if left to their own devices, so finish up by sewing a running stitch through the centre of each strand (in the same colour as the strand, so the stitches are invisible). When you come to an intersection, pick up a few stitches from the other strand, attaching the two strands together. Use the stitch markers to ensure that your intersections are properly aligned.

Remove the safety pins as you go, but don't remove the markers until you're confident the intersection is aligned and secured. When you're done, weave in the ends as you would with any crochet tails.

2.6 Rectangle Example Knot

The rectangle example knot is made up of two identical strands. Below are the stitching instructions, to be read as described in Section 2.4.

The finished dimensions are roughly 80 stitches wide by 30 stitches high. In crochet cotton, it is about the size of a bookmark; with medium yarn and a 4mm hook, it's a good size for a trivet.

If you work a swatch of 10 stitches by 6 rows, and measure the amount of yarn involved, the total yardage of this piece should be somewhere around 40 times that amount.

Figure 2.7: Rectangle Example Knot*

Make **two copies** of the strand below.

Foundation Chain: 10^{1T} / 3 / 7 / 3 / 10^{2B} / 6^{3T} / 2 / 6^{4B} / 3 / 7 / 3 / 6^{2T} / 2 / 6^{3B} / 10^{4T} / 3 / 7 / 3 / 10^{5B} / 10^{6T} / 10^{7B} / 3 / 15 / 3 / 3^{5T} / 3 / 15 / 3 / 3^{1B} / 3 / 15 / 3 / 10^{7T}

Row 1: 10^{7T} / 3 / 14 (-1) / 3 / 3^{1B} / 14 (-1) / 3 / 3^{5T} / 14 (-1) / 3 / 10^{7B} / 10^{6T} / 3^{5B} / 8 (+1) / 3 / 10^{4T} / 7^{3B} (+1) / 1 sc, 2 ch, 1 sc / 7 (+1) / 3^{2T} / 8 (+1) / 3 / 7^{4B} (+1) / 1 sc, 2 ch, 1 sc / 7 (+1) / 10^{3T} / 3^{2B} / 8 (+1) / 3 / 10^{1T}

Row 2: 10^{1T} / 3 / 10 (+2) / 3 / 10^{2B} / 7^{3T} / 2 sc, 2 ch, 2 sc / 7 / 3^{4B} / 10 (+2) / 3 / 7^{2T} / 2 sc, 2 ch, 2 sc / 7 / 10^{3B} / 3^{4T} / 10 (+2) / 3 / 10^{5B} / 10^{6T} / 3^{7B} / 12 (-2) / 3 / 3^{5T} / 12 (-2) / 3 / 3^{1B} / 12 (-2) / 3 / 10^{7T}

Row 3: 10^{7T} / 3 / 11 (-1) / 3 / 3^{1B} / 11 (-1) / 3 / 3^{5T} / 11 (-1) / 3 / 10^{7B} / 10^{6T} / 3^{5B} / 11 (+1) / 3 / 10^{4T} / 8^{3B} (+1) / 3 sc, 2 ch, 3 sc / 8 (+1) / 3^{2T} / 11 (+1) / 3 / 8^{4B} (+1) / 3 sc, 2 ch, 3 sc / 8 (+1) / 10^{3T} / 3^{2B} / 11 (+1) / 3 / 10^{1T}

Row 4: 10^{1T} / 3 / 12 (+1) / 3 / 10^{2B} / 9^{3T} (+1) / 4 sc, 2 ch, 4 sc / 9 (+1) / 3^{4B} / 12 (+1) / 3 / 9^{2T} (+1) / 4 sc, 2 ch, 4 sc / 9 (+1) / 10^{3B} / 3^{4T} / 12 (+1) / 3 / 10^{5B} / 10^{6T} / 3^{7B} / 10 (-1) / 3 / 3^{5T} / 10 (-1) / 3 / 3^{1B} / 10 (-1) / 3 / 10^{7T}

Row 5: 10^{7T} / 3 / 8 (-2) / 3 / 3^{1B} / 8 (-2) / 3 / 3^{5T} / 8 (-2) / 3 / 10^{7B} / 10^{6T} / 3^{5B} / 14 (+2) / 3 / 10^{4T} / 9^{3B} / 5 sc, 2 ch, 5 sc / 9 / 3^{2T} / 14 (+2) / 3 / 9^{4B} / 5 sc, 2 ch, 5 sc / 9 / 10^{3T} / 3^{2B} / 14 (+2) / 3 / 10^{1T}

Row 6: 10^{1T} / 3 / 15 (+1) / 3 / 10^{2B} / 10^{3T} (+1) / 6 sc, 2 ch, 6 sc / 10 (+1) / 3^{4B} / 15 (+1) / 3 / 10^{2T} (+1) / 6 sc, 2 ch, 6 sc / 10 (+1) / 10^{3B} / 3^{4T} / 15 (+1) / 3 / 10^{5B} / 10^{6T} / 3^{7B} / 7 (-1) / 3 / 3^{5T} / 7 (-1) / 3 / 3^{1B} / 7 (-1) / 3 / 10^{7T}

Finish Turn work. In outline colour, slip stitch across the last row. Then slip stitch across the foundation chain.

Assemble the two strands together, with crossings as in Figure 2.8 below. In the main colour yarn, sew a running stitch along each strand. When you come to an intersection, stitch through both layers.

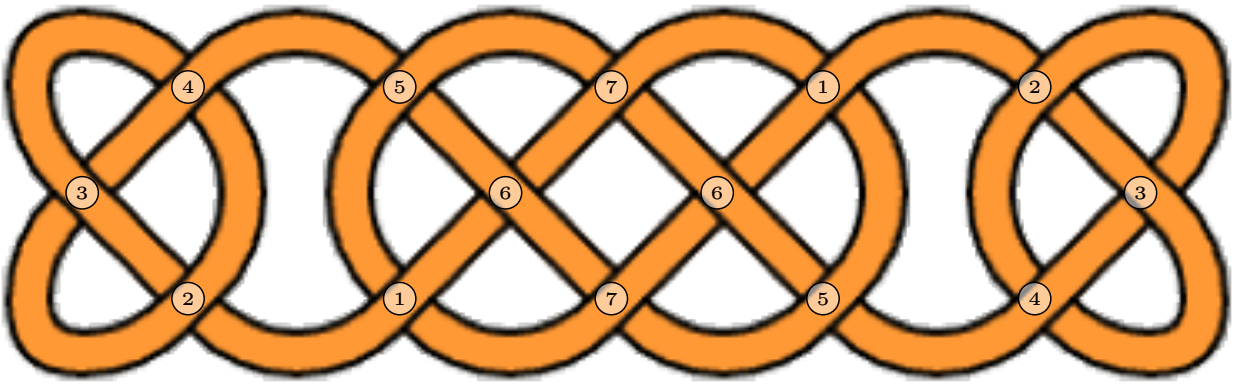
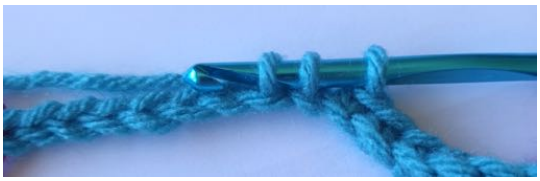


Figure 2.8: Intersection labels for rectangle example knot*

2.7 Photo Tutorial for Rectangle Example Knot

Figure 2.9: Lay out markers in order before starting foundation chain

Your first shaping stitches are decreases in Row 1. For Row 1, you can put the decreases and increases pretty much anywhere in a segment, since the whole thing starts out flat. In the figure below, we put one dec in the middle of the segment. Remember that decrease stitches are worked as slst2tog.



(a) Draw up a loop through two chains

(b) Bring the last loop on the hook through the other two to make a dec stitch

Figure 2.10: decrease stitch (slst2tog)

The first increase stitch of Row 1 comes when a segment of 7 chains should have 8 sc. The increase stitch is just 2 sc in the same stitch. Again, we don't have any shape yet to the segment, so we just plop the increase in the middle.

Figure 2.11: Increase stitch (2 sc in one st)

Later on in Row 1, a segment has the instructions (1 sc, 2 ch, 1 sc). This is the kind of increase that must be worked exactly where it's written. Note that you will not skip a stitch after the 2 chains.

(a) 2 ch worked; location of next stitch is indicated

(b) segment complete: 1 sc, 2 ch, 1 sc

Figure 2.12: Row 1: 1 sc, 2 ch, 1 sc

In Row 2, our first increase segment is written as 10 (+2). That means we need to have 2 increase stitches. In this example, we put our Row 1 increase from this segment right in the middle of the segment. Since we want to space out our increases along the segment, this time we'll put our increases at either end.

(a) In Row 1, the increase was placed in the middle of the segment

(b) In Row 2, we placed our increases at the edges of the segment. Notice how the strand is starting to form a smooth curve.

Figure 2.13: Row 2 increases

For the segment in Row 2 with instructions (2 sc, 2 ch, 2 sc): you'll be working into stitches from Row 1, which include both single crochets and chains. Do not work into the chain space, and do not skip any stitches.

(a) 2 ch worked; location of next stitch is indicated

(b) segment complete: 2 sc, 2 ch, 2 sc

Figure 2.14: Row 2: 2 sc, 2 ch, 2 sc

Below we show a segment in Row 3. The instructions say 11 (+1), so we need to place one increase stitch. The indicated part of the segment seems flatter that the other parts, so we place our increase there.

Figure 2.15: Flat part of curve in Row 3

When you've finished Row 6, break the main colour yarn, leaving a tail. Turn, and slip stitch across the entire strand.

(a) At end of Row 6, break main colour yarn

(b) Turn piece

(c) Bring up a loop of the outline colour

Figure 2.16: Setting up slip stitch outline

The first slip stitch may look different from the others; that will be fixed later, when we join the two ends of the strand.

You can choose to work the slip stitch outline through *back loops only* or through *both loops*. Working through both loops is sometimes more comfortable, but depending on your gauge, it may cause some of the main colour to peek past the outline.

(a) Both loops of sc

(b) Note the first slst looks vertical

(c) Slst through both loops

(d) Back loop of sc

(e) First slst slides towards back of work

(f) Slst through back loops only

Figure 2.17: Slip stitching through both loops vs. back loops only

Weave in the outline's starting tail as you go. Leave the main colour tail – this will be used later.

Figure 2.18: Back of work: weaving in outline tail, leaving main-colour tail free

For the outline slip-stitch rows, you don't need to flip the markers (but you can if you want).

You'll generally be putting one slst in each stitch. The exception is when you come to the points – the places where you worked 2 ch in each row. In order to keep the point sharp, you can add a chain stitch between your slip stitches.

(a) First main-colour chain of the point

(b) slst into chain (not chain space)

(c) ch 1

(d) do not skip any stitches

(e) slst into second main-colour chain

(f) sharp point

Figure 2.19: Slip stitching the outside of a point

When you come to the end of the row, cut your outline colour, leaving a 4-inch tail. Pull the yarn through the loop on your hook, but do not pull it tight.

(a) End of row

(b) Leave last loop somewhat loose

Figure 2.20: Ending slip-stitch outline on one edge of strand

The slip stitches appear on the right side of the work. Rotate the work 180 degrees (without turning to the wrong side) and slip-stitch along the foundation chain.

If you worked through both loops of the foundation chain in Row 1, then there is only one loop free for the outline. If you also worked through back loops only of Row 6, then both outlines will look symmetric on the back of the work.

(a) free loop of foundation chain

(b) pull up a loop of outline colour

(c) both outlines appear on the right side of the strand

Figure 2.21: Second slip-stitch outline

The first stitch may not quite make it to the very end of the strand (like in Figure 2.22c). At the end of the slip stitches, you can add a chain to help bridge the gap when you eventually close the strand and connect the outlines.

Generally, you will continue to put one slip stitch into each stitch. When you come to the inside of a point, you can slst2tog to keep the curve tight.

(a) Inside of a point

(b) pull up a loop through two innermost stitches

(c) pull last loop through the other two

Figure 2.22: Slip-stitch outline on inside of a point

(a) At the end of the row, you might want to make an extra chain, if there was a gap at the start of the outline

(b) Leave a tail and pull through

Figure 2.23: Ending the second slip-stitch outline

Figure 2.24: First strand finished

Make the second strand in just the same way as the first. If you like, you can pull out the unlabeled stitch markers to re-use, but leave in the labeled ones.

When you've finished both strands, it's time to assemble them. The diagram in Figure 2.8 is your guide. Line up labelled markers with the same number. "T" goes on top, "B" on the bottom. The markers should be perpendicular to one another.

If you haven't already, remove the unlabelled stitch markers – their use was only for placing increases and decreases, which is over.

Figure 2.25: Intersection crossing

It's convenient to start the weaving with the pointy parts of the two strands. Weave and pin these intersections (numbered 2, 3, and 4).

Figure 2.26: Starting weaving with the pointy parts

Now weave the two strands together, matching up intersections and pinning them in place.

(a) Line up two strands

(b) Weave together

Figure 2.27: Two strands woven together

At the intersection numbered 6, the top strand has a marker, while the bottom strand is broken. (The location of this break serves as your marker for aligning the intersection.) You'll join the strand at these breaks, on the back side of the work.

Figure 2.28: Back of work, with join locations indicated

Using the main colour tails, sew the strands together at the joins. I like to leave the tail-weaving for after, since the tail can serve to mark the location of the join. The seam should be aligned perpendicularly to the running stitch marker 6T.

(a) Thread tail into tapestry needle

(b) Sew together

Figure 2.29: Two ends of a strand joined together

The outlines are broken, so join these as well. The front of the strand is hidden because it's on the bottom of a join, but you can work an invisible join if you're feeling fancy.

(a) Pull back the top strand to see the right side of the bottom strand.

(b) Thread the tail of the outline colour into a tapestry needle. Pass the tail underneath the first stitch on the far side of the join

(c) Bring the tail back down through the last stitch, where you had pulled it through when you broke the yarn

(d) You should now have a loop that looks like the other slip stitches, but looser

(e) Pull the loop until it is the same size as the other stitches. Fanciness achieved – the border should look unbroken.

Figure 2.30: Invisible Join

On the back of the work, make a few stitches with the tail to make the back of the outline continuous as well.

(a) Pink outline has a gap

(b) Gap stitched over

Figure 2.31: Joining the outline on the back side

If you ended things now, your knot would be closed, but the strands might slide around. To secure them, the last step is to work a running stitch along the centre of both strands.

Cut a length of yarn of the main colour, 1.5 to 2 times the length of the strand. That's a lot of yarn to sew with, so to make things easier, when you make your first stitch, pull half the yarn through. Now you can stitch with each half separately.

(a) Half of yarn pulled through the first stitch

(b) Running stitch with one side of the yarn

Figure 2.32: Start running stitch

When you come to an intersection, pick up yarn from the intersecting strand to fasten it in place. This is where you finalize the alignment of all your intersections, so it's worth being careful. Fold back the top strand to make sure the two markers align at their middles, perpendicularly.

Figure 2.33: Checking intersection alignment

After stitching along the entire strand, weave in the tails as usual. Once both strands have been stitched, remove all markers, weave in all ends, and block if desired.

(a) Front

(b) Back

Figure 2.34: Finished piece

2.8 Border Example Knot

This knot is made up of repeating segments, so it can be worked as long as you like. Its finished height is about 30 stitches.

It could make a table runner, or decorative lace to sew to a hem. Because the edges of the knot are open, it twists easily, making it a difficult choice for a scarf or any other item that isn't usually laying flat.

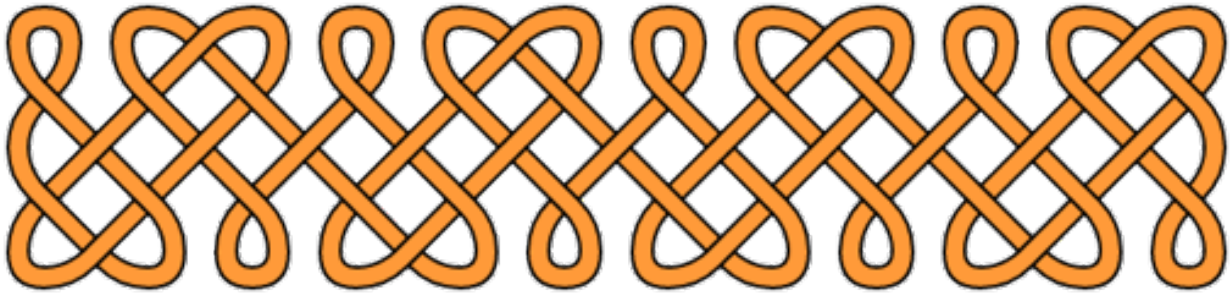


Figure 2.35: Border Example Knot*

End Parts: Make 2

Foundation Chain: 10 $\overset{3T}{/}$ 6 / 2 / 6 $\overset{2B}{/}$ 10 $\overset{1T}{/}$ 3 / 7 / 3 $\overset{7B}{/}$ 10 / 17 / 10 $\overset{7T}{/}$ 10

Row 1: 10 $\overset{7T}{/}$ 9 (-1) / 14 (-3) / 9 (-1) $\overset{7B}{/}$ 3 / 8 (+1) / 3 $\overset{1T}{/}$ 10 $\overset{2B}{/}$ 7 (+1) / 1 sc, 2 ch, 1 sc / 7 (+1) $\overset{3T}{/}$ 10

Row 2: 10 $\overset{3T}{/}$ 7 / 2 sc, 2 ch, 2 sc / 7 $\overset{2B}{/}$ 10 $\overset{1T}{/}$ 3 / 10 (+2) / 3 $\overset{7B}{/}$ 9 / 12 (-2) / 9 $\overset{7T}{/}$ 10

Row 3: 10 $\overset{7T}{/}$ 8 (-1) / 9 (-3) / 8 (-1) $\overset{7B}{/}$ 3 / 11 (+1) / 3 $\overset{1T}{/}$ 10 $\overset{2B}{/}$ 8 (+1) / 3 sc, 2 ch, 3 sc / 8 (+1) $\overset{3T}{/}$ 10

Row 4: 10 $\overset{3T}{/}$ 9 (+1) / 4 sc, 2 ch, 4 sc / 9 (+1) $\overset{2B}{/}$ 10 $\overset{1T}{/}$ 3 / 12 (+1) / 3 $\overset{7B}{/}$ 7 (-1) / 7 (-2) / 7 (-1) $\overset{7T}{/}$ 10

Row 5: 10 $\overset{7T}{/}$ 7 / 4 (-3) / 7 $\overset{7B}{/}$ 3 / 14 (+2) / 3 $\overset{1T}{/}$ 10 $\overset{2B}{/}$ 9 / 5 sc, 2 ch, 5 sc / 9 $\overset{3T}{/}$ 10

Row 6: 10 $\overset{3T}{/}$ 10 (+1) / 6 sc, 2 ch, 6 sc / 10 (+1) $\overset{2B}{/}$ 10 $\overset{1T}{/}$ 3 / 15 (+1) / 3 $\overset{7B}{/}$ 6 (-1) / 2 (-2) / 6 (-1) $\overset{7T}{/}$ 10

Finish Turn and slip stitch along the last row in the outline colour; slip stitch along the foundation chain in the outline colour as well.

Oval Part: make N

Make as many as you need to reach your desired length.

Foundation Chain: 10 $\overset{2T}{/}$ 6 / 2 / 6 $\overset{1B}{/}$ 10 $\overset{4T}{/}$ 10 $\overset{3B}{/}$ 10 $\overset{2T}{/}$ 6 / 2 / 6 $\overset{1B}{/}$ 10 $\overset{4T}{/}$ 10

Row 1: 10 $\overset{4T}{/}$ 10 $\overset{1B}{/}$ 7 (+1) / 1 sc, 2 ch, 1 sc / 7 (+1) $\overset{2T}{/}$ 10 $\overset{3B}{/}$ 10 $\overset{4T}{/}$ 10 $\overset{1B}{/}$ 7 (+1) / 1 sc, 2 ch, 1 sc / 7 (+1) $\overset{2T}{/}$ 10

Row 2: 10 $\overset{2T}{/}$ 7 / 2 sc, 2 ch, 2 sc / 7 $\overset{1B}{/}$ 10 $\overset{4T}{/}$ 10 $\overset{3B}{/}$ 10 $\overset{2T}{/}$ 7 / 2 sc, 2 ch, 2 sc / 7 $\overset{1B}{/}$ 10 $\overset{4T}{/}$ 10

Row 3: 10 $\overset{4T}{/}$ 10 $\overset{1B}{/}$ 8 (+1) / 3 sc, 2 ch, 3 sc / 8 (+1) $\overset{2T}{/}$ 10 $\overset{3B}{/}$ 10 $\overset{4T}{/}$ 10 $\overset{1B}{/}$ 8 (+1) / 3 sc, 2 ch, 3 sc / 8 (+1) $\overset{2T}{/}$ 10

Row 4: 10 $\overset{2T}{/}$ 9 (+1) / 4 sc, 2 ch, 4 sc / 9 (+1) $\overset{1B}{/}$ 10 $\overset{4T}{/}$ 10 $\overset{3B}{/}$ 10 $\overset{2T}{/}$ 9 (+1) / 4 sc, 2 ch, 4 sc / 9 (+1) $\overset{1B}{/}$ 10 $\overset{4T}{/}$ 10

Row 5: 10 $\overset{4T}{/}$ 10 $\overset{1B}{/}$ 9 / 5 sc, 2 ch, 5 sc / 9 $\overset{2T}{/}$ 10 $\overset{3B}{/}$ 10 $\overset{4T}{/}$ 10 $\overset{1B}{/}$ 9 / 5 sc, 2 ch, 5 sc / 9 $\overset{2T}{/}$ 10

Row 6: 10 $\overset{2T}{/}$ 10 (+1) / 6 sc, 2 ch, 6 sc / 10 (+1) $\overset{1B}{/}$ 10 $\overset{4T}{/}$ 10 $\overset{3B}{/}$ 10 $\overset{2T}{/}$ 10 (+1) / 6 sc, 2 ch, 6 sc / 10 (+1) $\overset{1B}{/}$ 10 $\overset{4T}{/}$ 10

Finish Turn and slip stitch along the last row in the outline colour; slip stitch along the foundation chain in the outline colour as well.

Twisty Part: make $N - 1$

Make one fewer than you made of the oval part.

Foundation Chain: 10 ^{1T} / 10 ^{2B} / 10 / 14 / 10 ^{3T} / 10 ^{4B} / 10 ^{5T} / 6 / 2 / 6 ^{5B} / 10 ^{6T} / 10 ^{5B} / 10 / 17 / 10 ^{5T} / 10 ^{4B} / 10 ^{3T} / 6 / 2 / 6 ^{2B} / 10 ^{1T} / 10

Row 1: 10 ^{1T} / 10 ^{2B} / 7 (+1) / 1 sc, 2 ch, 1 sc / 7 (+1) ^{3T} / 10 ^{4B} / 10 ^{5T} / 9 (-1) / 14 (-3) / 9 (-1) ^{5B} / 10 ^{6T} / 10 ^{5B} / 7 (+1) / 4 (+2) / 7 (+1) ^{5T} / 10 ^{4B} / 10 ^{3T} / 9 (-1) / 5 sc, 2 dec sts, 5 sc / 9 (-1) ^{2B} / 10 ^{1T} / 10

Row 2: 10 ^{1T} / 10 ^{2B} / 9 / 4 sc, 2 dec sts, 4 sc / 9 ^{3T} / 10 ^{4B} / 10 ^{5T} / 7 / 7 (+3) / 7 ^{5B} / 10 ^{6T} / 10 ^{5B} / 9 / 12 (-2) / 9 ^{5T} / 10 ^{4B} / 10 ^{3T} / 7 / 2 sc, 2 ch, 2 sc / 7 ^{2B} / 10 ^{1T} / 10

Row 3: 10 ^{1T} / 10 ^{2B} / 8 (+1) / 3 sc, 2 ch, 3 sc / 8 (+1) ^{3T} / 10 ^{4B} / 10 ^{5T} / 8 (-1) / 9 (-3) / 8 (-1) ^{5B} / 10 ^{6T} / 10 ^{5B} / 8 (+1) / 9 (+2) / 8 (+1) ^{5T} / 10 ^{4B} / 10 ^{3T} / 8 (-1) / 3 sc, 2 dec sts, 3 sc / 8 (-1) ^{2B} / 10 ^{1T} / 10

Row 4: 10 ^{1T} / 10 ^{2B} / 7 (-1) / 2 sc, 2 dec sts, 2 sc / 7 (-1) ^{3T} / 10 ^{4B} / 10 ^{5T} / 9 (+1) / 12 (+3) / 9 (+1) ^{5B} / 10 ^{6T} / 10 ^{5B} / 7 (-1) / 7 (-2) / 7 (-1) ^{5T} / 10 ^{4B} / 10 ^{3T} / 9 (+1) / 4 sc, 2 ch, 4 sc / 9 (+1) ^{2B} / 10 ^{1T} / 10

Row 5: 10 ^{1T} / 10 ^{2B} / 9 / 5 sc, 2 ch, 5 sc / 9 ^{3T} / 10 ^{4B} / 10 ^{5T} / 7 / 4 (-3) / 7 ^{5B} / 10 ^{6T} / 10 ^{5B} / 9 / 14 (+2) / 9 ^{5T} / 10 ^{4B} / 10 ^{3T} / 7 / 1 sc, 2 dec sts, 1 sc / 7 ^{2B} / 10 ^{1T} / 10

Row 6: 10 ^{1T} / 10 ^{2B} / 6 (-1) / 0 sc, 2 dec sts, 0 sc / 6 (-1) ^{3T} / 10 ^{4B} / 10 ^{5T} / 10 (+1) / 17 (+3) / 10 (+1) ^{5B} / 10 ^{6T} / 10 ^{5B} / 6 (-1) / 2 (-2) / 6 (-1) ^{5T} / 10 ^{4B} / 10 ^{3T} / 10 (+1) / 6 sc, 2 ch, 6 sc / 10 (+1) ^{2B} / 10 ^{1T} / 10

Finish Turn and slip stitch along the last row in the outline colour; slip stitch along the foundation chain in the outline colour as well.

Weave together the knot according to the labels below.

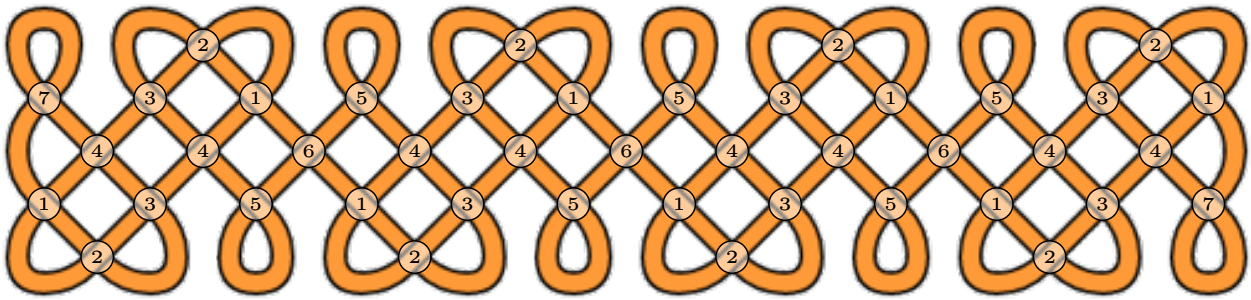


Figure 2.36: Intersection labels for border example knot*

Sew through the strands with a running stitch in the main colour to secure the intersections.

For more details on the process, see the phototutorial for the rectangle example knot, Section 2.7.

2.9 Example knot specifications

The strands' patterns and markers are given below, formatted for the program from Section 4.6.2. If you'd like to use the Python program to change some parameters of the knots, you can copy-paste the lists.

Rectangle Example Knot:

```
pattern=["S","IQ","S","IP","IQ","IP","S","IQ","S","S","OQ","OQ","OQ","S"]
markers=["1T","2B","3T","4B","2T","3B","4T","5B","6T","7B","5T","1B","7T"]
```

Border Example Knot:

End Parts:

```
pattern=["S","IP","S","IQ","OT","S"]
markers=["3T","2B","1T","7B","7T"]
```

Oval Parts:

```
pattern=["S","IP","S","S","S","IP","S","S",]
markers=["2T","1B","4T","3B","2T","1B","4T"]
```

Twisty Parts:

```
pattern=["S","S","OP","S","S","IT","S","S","OT","S","S","IP","S","S"]
markers=["1T","2B","3T","4B","5T","5B","6T","5B","5T","4B","3T","2B","1T"]
```

Chapter 3

Drawing Knots

3.1 Algorithmic Approach to Drawing Rectangular Knots

Although many shapes of knot exist, we will deal only with **rectangular** knots. The ideas in Chapter 5, however, could be adapted to knots of other regular shapes.

The website https://w-shadow.com/celtic-knots/ (which requires a "fairly modern browser," according to their own documentation) is a convenient tool to generate knots in your browser. To start playing around with it, click a red or blue dot in the grid, move your mouse to an adjacent dot of the same colour, and click it.

I have no affiliation with the linked website. If it breaks, or goes behind a paywall, please message me on Ravelry so I can look for an alternative resource.

An explanation of drawing these knots by hand, using the same technique as the website, is given below. This will help you understand the terms in Chapters 4 and 5.

We start with a rectangular grid of dots, and highlight every other one, as in Figure 3.1. The corners of the rectangle should be unhighlighted. (The requires an odd number of starting dots for both the length and the width.)

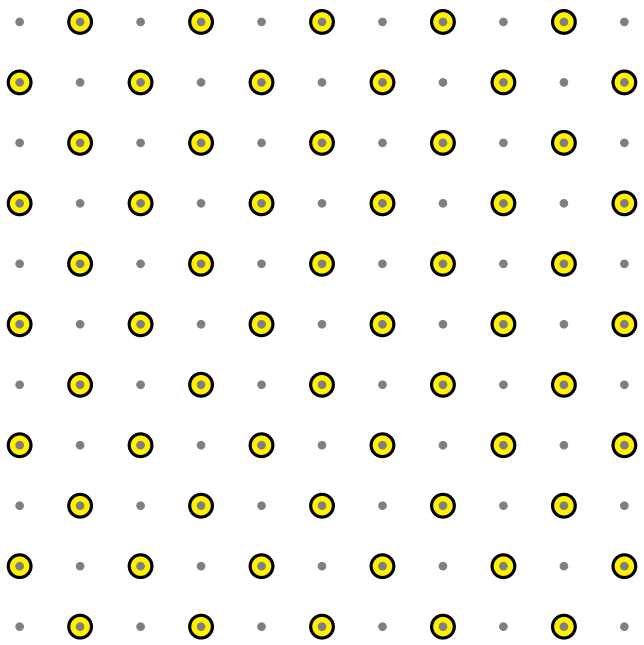


Figure 3.1: Blank Starting Grid

If we have a totally blank knot – basically just a weave– the strands will travel diagonally across the grid, crossing at the highlighted dots (Figure 3.2). Note how the strands move between highlighted dots that are diagonally adjacent.

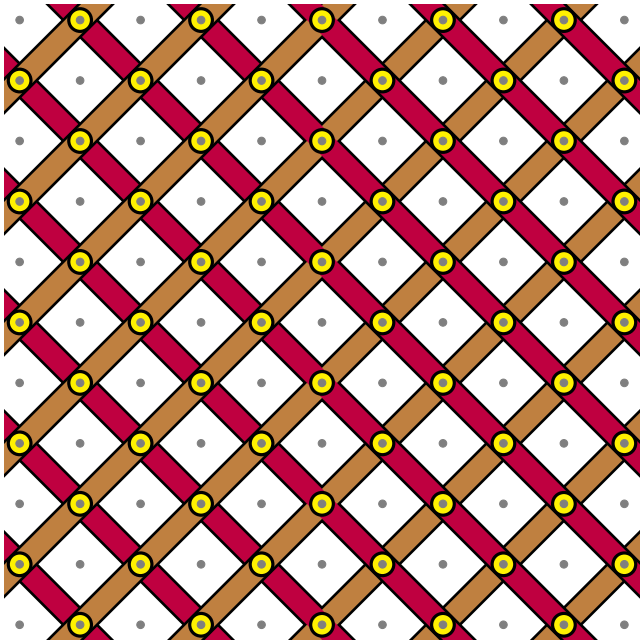
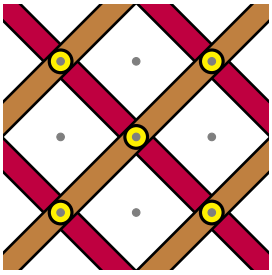


Figure 3.2: Blank Knot

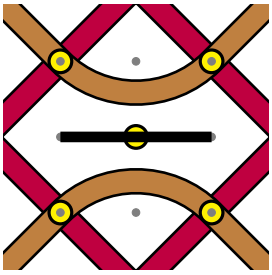
To add features to the knot, draw vertical or horizontal lines at the highlighted dots, stretching from one background dot to another. At these lines, the usual crossing is broken, and replaced with two curves. We'll

call these lines feature walls.

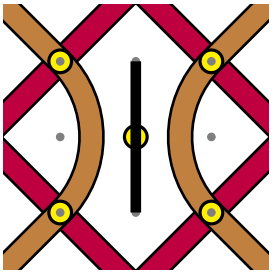
If you're drawing these by hand, the following mental image might be useful: at every dot, the strand would like to travel to the next dot diagonally. When it hits one of these lines, that's a wall, deflecting its path.



(a) Centre dot unmarked



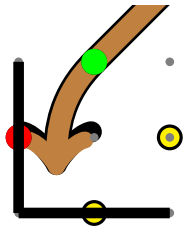
(b) Centre dot with horizontal feature wall



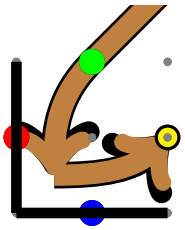
(c) Centre dot with vertical feature wall

Figure 3.3: A single feature wall causes the strand to make a quarter turn

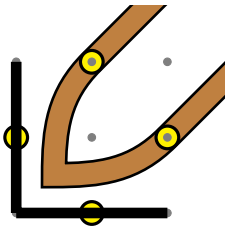
We want our knot to not wander off[1] the picture, so let's add these feature walls around the border of the blank knot. We come into a new situation at the corners: two feature walls that touch each other. Feature walls should never *cross* one another, but they might meet at their ends. When this happens, we keep our mental image of the feature walls as walls that deflect the knot strands. This causes our strands to make a corner.



(a) At green dot, strand wants to move to next dot, in red. Since there is a feature wall there, the strand is deflected.



(b) From the red deflection, the strand wants to move to the blue dot. Since there is a feature wall there, it is deflected again.



(c) The two feature walls meeting at a corner cause the strand to make a point

Figure 3.4: Two feature walls making a point

All together, our blank knot looks like this:

[1] For different applications, you actually might want to leave loose ends. For example, if you wanted the knot to be an insert, and connect the strand to a larger crochet piece. In our examples we close off the edges of the grid for simplicity, not necessity. If you understand the method as described, you'll have no problem using open edges instead.

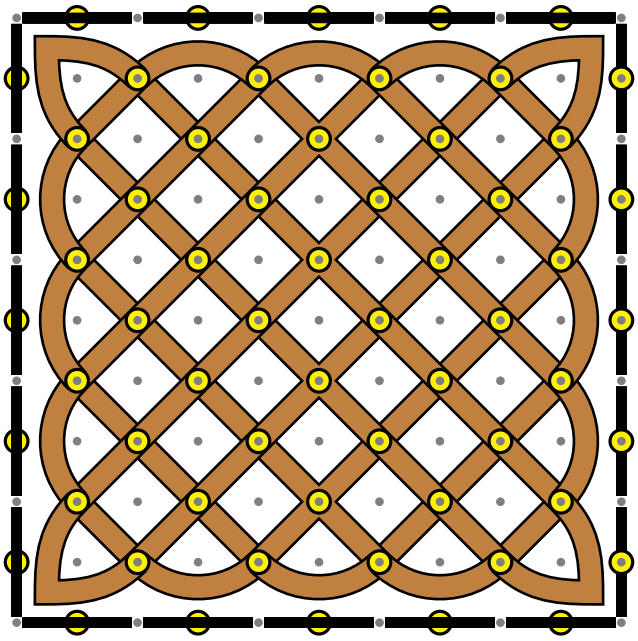
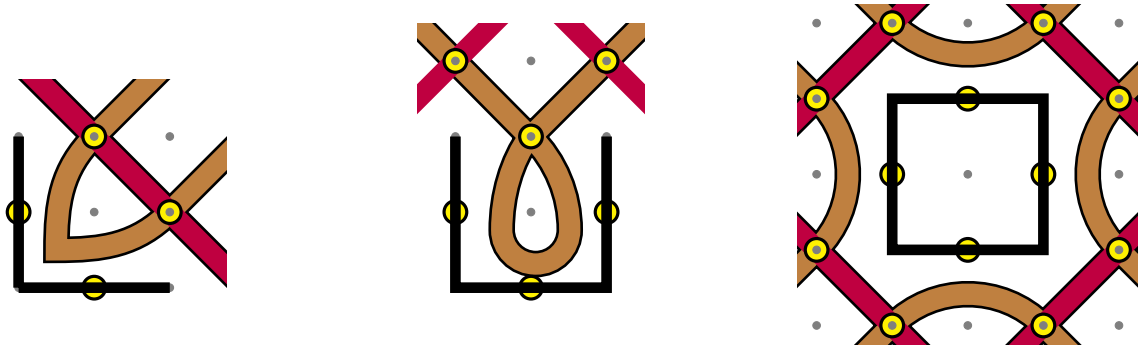


Figure 3.5: Blank Knot with Borders

From here, you can experiment with adding feature walls in the knot's interios. For example, it's possible to have three feature walls make a dead end. This makes a self-twist in the strand.



(a) Corner Point (b) Twist (c) Hole

Figure 3.6: Combinations of feature walls

If you add adjacent rows of parallel feature walls, you'll get a run – a long straight horizontal or vertical piece. This case is significantly more complicated than the others, so we've included it at the end of this chapter, in 3.3.

After your feature walls are all added, add in the crossings. At every crossing, one strand will go on top, and the other on the bottom. The hard way of doing this is following a strand and keeping track of over/under for yourself. The easy way is to recognize that all crossings in a row or column have the same crossing orientation: in all of them, the top strand has the same orientation (bottom left to top right, or bottom right to top left).

The knot in Figure 3.7 has no internal feature walls, but even with feature walls, each row or column will have the same crossing orientations.

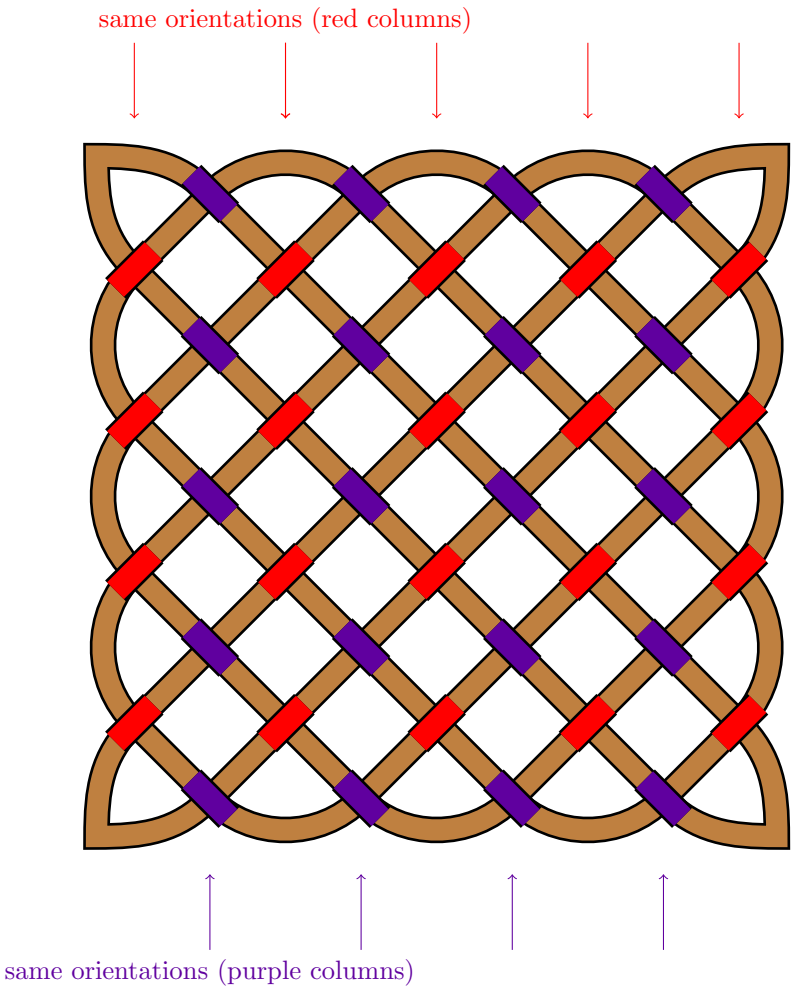


Figure 3.7: Knot with oriented crossings

Although this method requires the grid to be rectilinear (i.e. made out of horizontal and vertical lines), it doesn't have to actually be a rectangle. You can block out different shapes from an underlying rectangular grid.

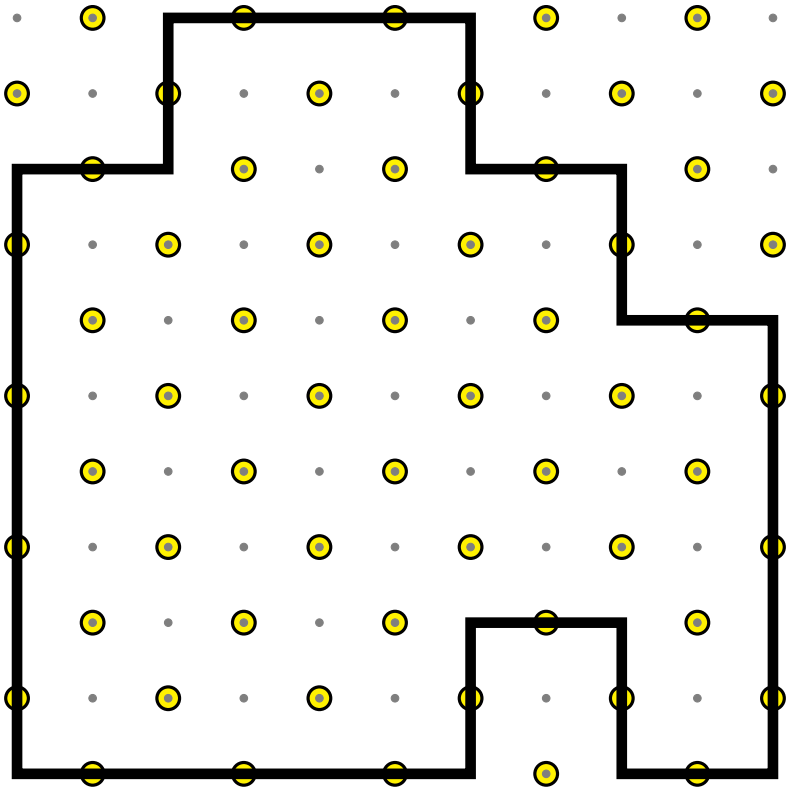


Figure 3.8: Alternate Grid Shape

3.2 Practical Considerations

For knots that you will turn into physical objects, the end use of the object should be taken into consideration. For example, anything that won't spend its life laying down, or tacked to something else, probably shouldn't have any large continuous gaps. So you might want to stay away from long strings of feature walls that make big slits in the pattern.

Similarly, a twist caused by three feature walls making a dead end (as in Figure 3.6b) will lead to a flappy bit. This is probably also best reserved for pieces that will lay flat.

Something else to consider in your finished pattern is how many separate strands make it up. Follow a strand around, possibly colouring it, until you return to your starting location. Longer strands are somewhat harder to arrange when it comes time to assemble your piece. Not impossible, surely, but keep in mind you'll want to be extra careful labelling your intersections. If you do want to use a pattern with a very long strand, consider breaking the strand into several pieces and sewing them together.

3.3 Special Case: Drawing Runs

When you add parallel adjacent feature walls, the strand turns into a vertical or horizontal straight line, called a run. That straight part is easy enough to draw. The hard part comes when the run *ends*. The easiest way to draw these is to keep in mind the "deflecting" paradigm, that strands want to go to their diagonally-adjacent dot, but are deflected when they run into a feature wall. The exposition below goes into greater detail about the various cases, but if you're comfortable drawing using the deflection model, you shouldn't need this level of detail for knot drawing. (You *will* need this level of detail for translating a knot that contains a run into a crochet pattern.)

In the simplest case, the run ends directly at an intersection. That is, the strand goes from the run directly to a dot. For this case, draw a smooth arc from the end of the run to the dot. The strand should make a 45-degree turn, so it's back in its usual diagonal orientation when it hits the dot.

(a) Run using an odd number of feature walls (seven)

(b) Run using an even number of feature walls (six)

Figure 3.9: D-type corners

We call this type of corner D-type, where D stands for "direct." If the first dot after a run has a feature wall, then the corner will be of a different type.

(a) End dot has no wall: D-type (see Figure 3.9)

(b) End dot has perpendicular wall: not D-type

Figure 3.10: End of a Run

If the red dots in Figure 3.10 had a horizontal feature wall, they would just be a part of the run. So the end dot of a run will not have a feature wall parallel to the run. It can either have no feature wall (and make a D-type corner, as in Figure 3.9), or a feature wall perpendicular to the direction of the run (in the cases pictures above, vertical).

In the situation shown in Figure 3.10b, the strand will be "deflected" downwards by the red wall. In this case, we turn our attention to the next dot the strand wants to go to, shown in green below.

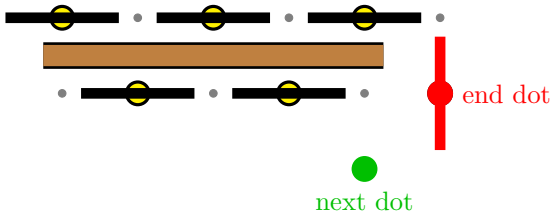
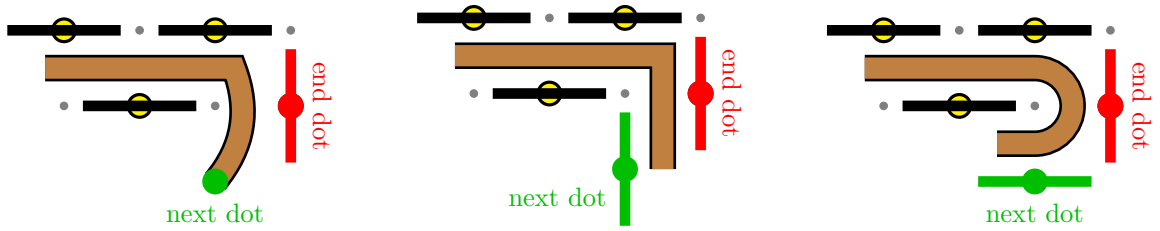


Figure 3.11: Next dot at the end of a run

The green dot can have no wall, a wall parallel to the run, or a wall perpendicular to the run.



(a) next dot has no feature wall: J-type

(b) next dot has perpendicular feature wall: L-type

(c) next dot has parallel feature wall: C-type

Figure 3.12: J-type, L-type, and C-type corners

If the next dot has no feature wall (Figure 3.12a), make a corner at "end dot," then a smooth arc from the run to the green dot. You are now at an intersection, and the component has ended. Since the strand makes a turn that looks like the letter J, we call this a J-type corner.

If the next dot has a feature wall perpendicular to the original run (parallel to the wall on the end dot, as shown in Figure 3.12b), then the strand makes a right-angle turn. It is now starting a new run, perpendicular to the last. There has been no crossing, so you are still in the same component. Since the strand makes an L-shape, we call this an L-type corner.

If the next dot has a feature wall parallel to the original run (perpendicular to the wall on the end dot, as shown in Figure 3.12c), the strand makes a half-circle. It is now starting a new run, parallel to the last. There has been no crossing, so you are still in the same component. Since the strand makes a C-shape turn, we call this a C-type corner.

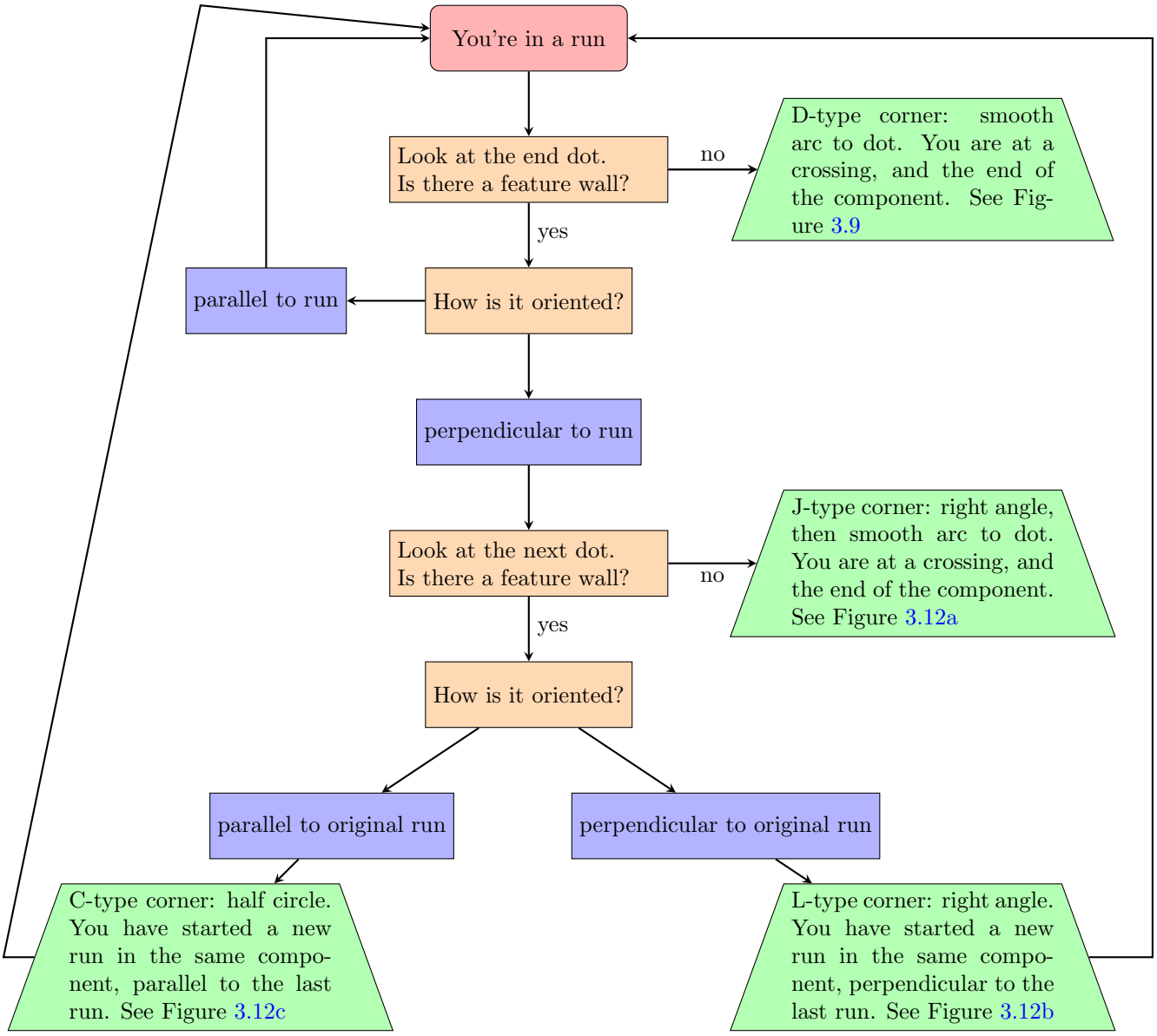


Figure 3.13: An entirely unnecessary flowchart for drawing runs

Chapter 4

Translating Knots to Crochet, Part 1

Now that you've drawn your knot, it's time to translate it into a crochet pattern. The assembly will follow the steps in the example knots, Section 2.5.

4.1 Starting location and direction

Choose a location on a strand where it is on the bottom of a crossing. That way, the join will be hidden when the knot is assembled. If possible, also choose an under-crossing between two straight components. It's a little easier to start and end without worrying about negotiating a curve.

You might want to delete a stitch from one or both ends of your strand. That way, when the two ends are joined together, the extra bulk from the join will not add unwanted length.

If you plan on slip-stitching an outline before the piece is assembled, then you'll need to know which sides are right and wrong. I like to orient my starting position so that the foundation-chain side of the strand is at the bottom, then go left-to-right (Figure 4.1). As a right-handed crocheter, I work right-to-left. So the foundation chain, and even-numbered rows, are worked with the wrong side up. After the last row (in the examples, Row 6, but it is the same for any even number of rows) we turn the work to slip stitch with the right side up.

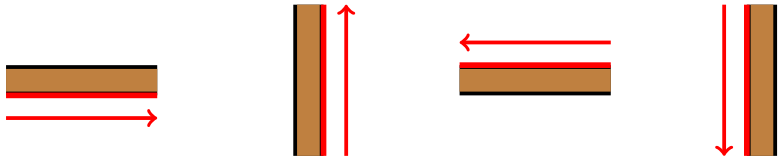


Figure 4.1: Foundation chain shown in red, with left-to-right direction along strand indicated by arrow

The next step is to follow the curve around and identify each component. Components are separated by crossings.

4.2 Standard Components

There are four standard components, each of which starts and ends at a dot. A section directly between two dots, with no feature walls, is straight. A single feature wall not touching any other feature walls causes a quarter turn (Figure 3.3b). Two feature walls meeting at a corner makes a point (Figure 3.6a), and three feature walls in a U-shape make a twist (Figure 3.6b).

Once you've identified the location on a strand where you want to start, draw a line on one edge. That will be the edge of the foundation row. Follow the segment around until you end up where you started, identifying each component you pass. Components are separated by crossings. That is, they will start and stop at dots. Note whether the edge you're following is on the inside or outside of a component (does not apply to straight components).

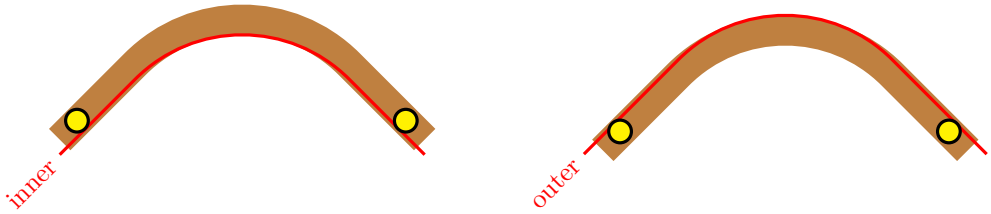


Figure 4.2: Quarter Turn

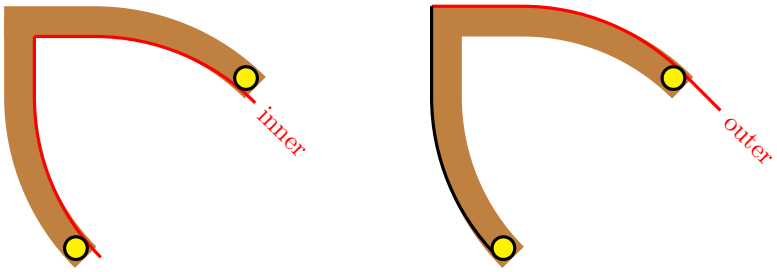


Figure 4.3: Point

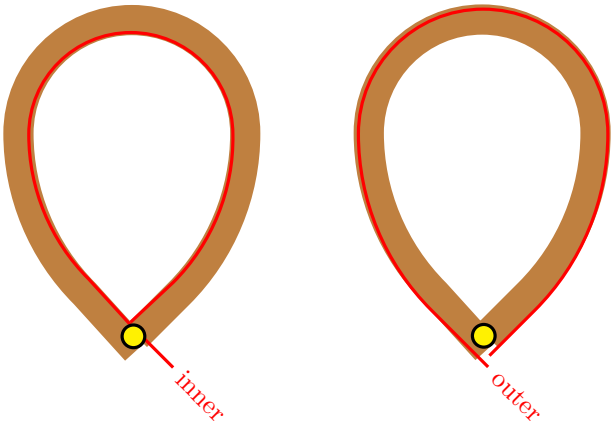


Figure 4.4: Twist

4.3 Example: Identifying Standard Components

By way of example, we will identify the components of the border example knot. Below we show this knot with all its interior highlighted dots and feature walls.

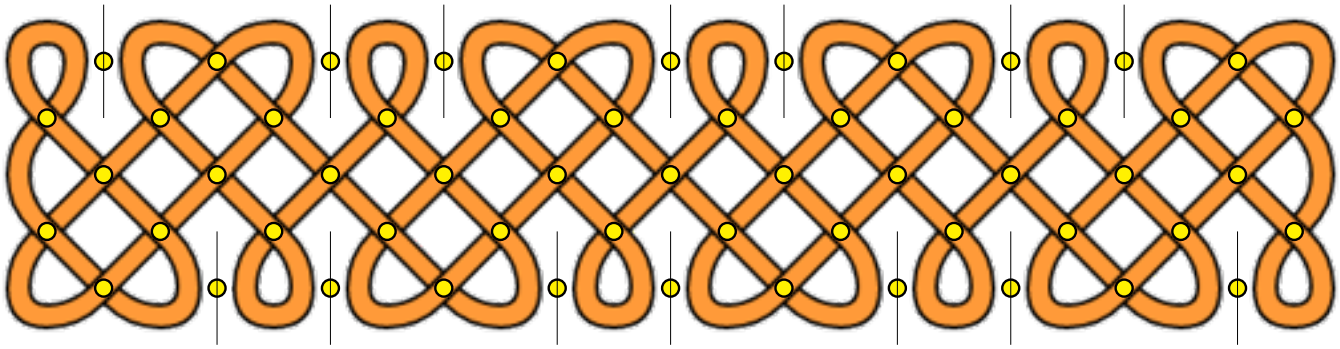


Figure 4.5: border example knot with dots shown*

A component will follow the strand from one dot to the next. Let's start at the far left of the knot, shown in purple below. Ideally, we like to start at under-crossings where two straight segments meet. So, we start at the red dot.

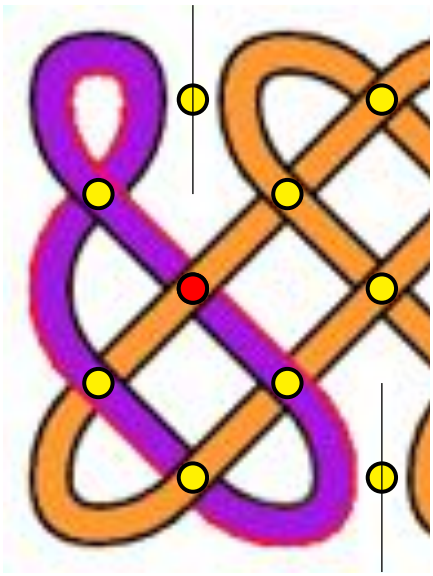


Figure 4.6: End piece of border example knot*

Starting bottom-to-top along the red line, our components are as follows: straight, inner twist, outer quarter, straight, outer point, straight.

If we were to follow the black edge instead, now starting top-to-bottom, our components would be: straight, inner point, straight, inner quarter, outer twist, straight.

Let's move on to the oval part, shown in purple below. Ideally, we like to start at under-crossings where two straight segments meet. So, we start at the red dot.

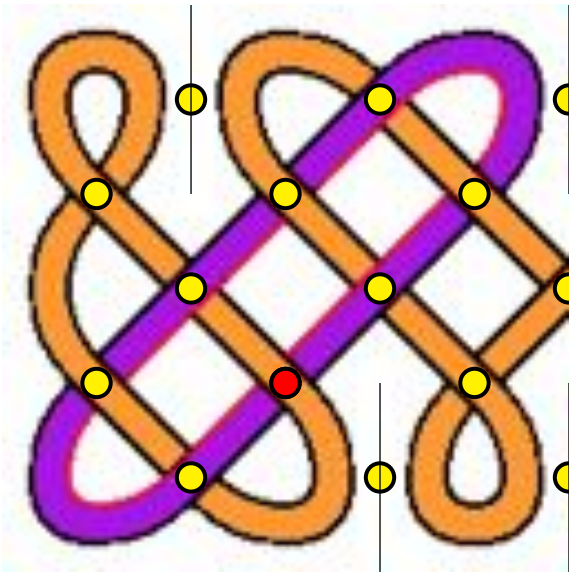


Figure 4.7: Oval part of border example knot*

Starting top-to-bottom, our components are as follows: straight, inner point, straight, straight, straight, inner point, straight, straight.

If we were to follow the black edge instead, now starting bottom-to-top, our components would be: straight, straight, outer point, straight, straight, straight, outer point, straight.

Finally, we have the twisted shape, shown in purple below. Ideally, we like to start at under-crossings where two straight segments meet. So, we start at the red dot.

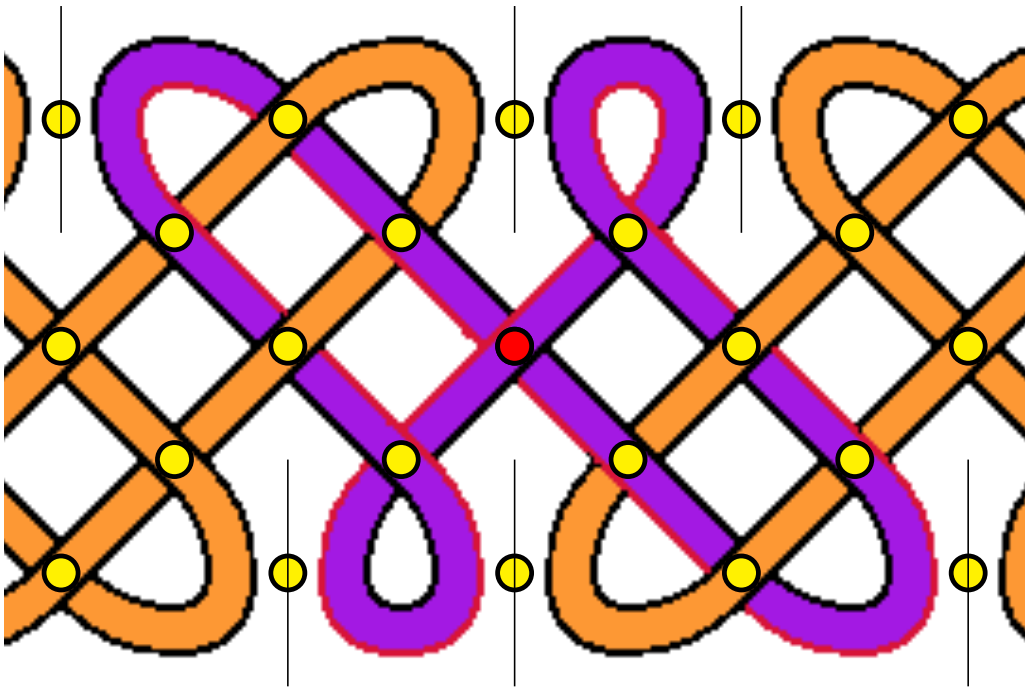


Figure 4.8: Twisted piece of border example knot*

Starting top-to-bottom along the red edge, our components are as follows: straight, straight, outer point, straight, straight, inner twist, straight, straight, outer twist, straight, straight, inner point, straight, straight.

If we were to follow the black edge instead, still moving bottom-to-top, our components would be: straight, straight, outer point, straight, straight, inner twist, straight, straight, outer twist, straight, straight, inner point, straight, straight.

4.4 Example: Identifying Runs

When you add *parallel, adjacent feature walls*, the situation gets a lot more complicated. Adjacent feature walls cause long straight parts, which we'll call runs. Runs start and end at corners. A corner can be one of four types. For identifying the types of corners, see 3.3. The length of a run is the number of dots involved in the parallel lines of feature walls, as in Figure 3.9.

For identifying corner types, remember that C-type, J-type, and L-type corners are named the way they look. D-type corners are "direct" corners, where the first dot that isn't on the run has no feature wall. Components with runs start and end at D-type or J-type corners, while C-type and L-type corners connect two runs inside the same component.

For practice, we'll identify the corner types and run lengths in the runs of the knot in Figure 4.9.

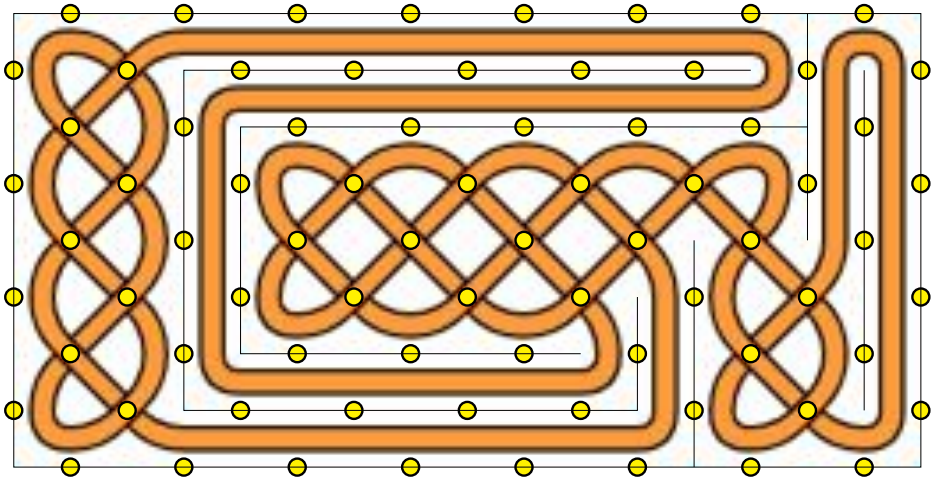


Figure 4.9: Example: identifying runs*

Let's start with the top, horizontal run, marked in purple below. The adjacent parallel lines of feature walls that define the straight section are highlighted in red.

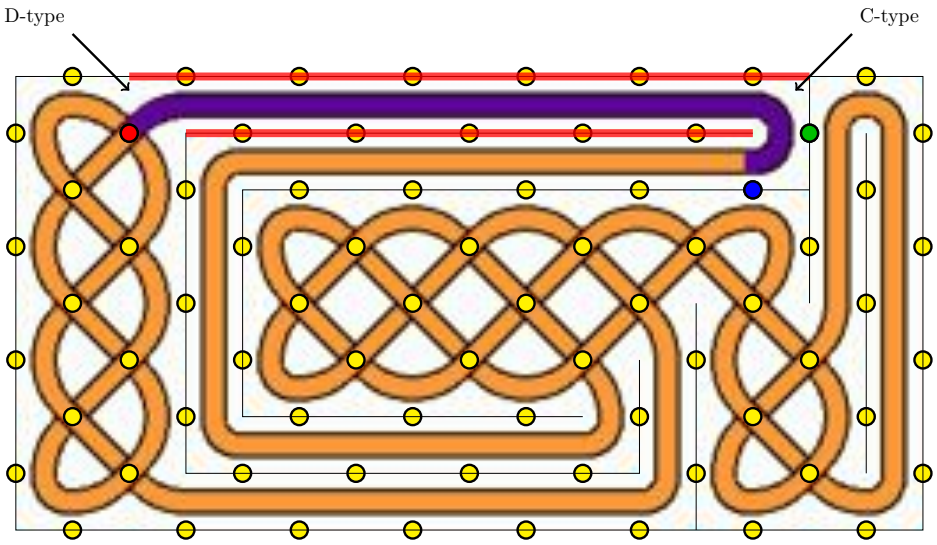


Figure 4.10: D-type and C-type corners*

Since there are eleven dots in the walls (6 on top and 5 below), the length of this run is **11**.

On the left, the red dot is the first dot that is off the run. (If it had a horizontal feature wall, it would extend the run.) Since this dot has no feature wall, that corner is **D(irect)-type**.

On the right, the green dot is the first dot that is off the run. (If it had a horizontal feature wall, it would extend the run.) This dot has a feature wall perpendicular to the original run, connecting to a feature wall (through the blue dot) parallel to the original feature wall. So the right corner of the run is **C-type**. Notice how the run makes a 180 degree (C-shaped) turn into a new run.

Let's continue to the next run, marked in purple below. The adjacent parallel lines of feature walls that define the straight section are highlighted in red.

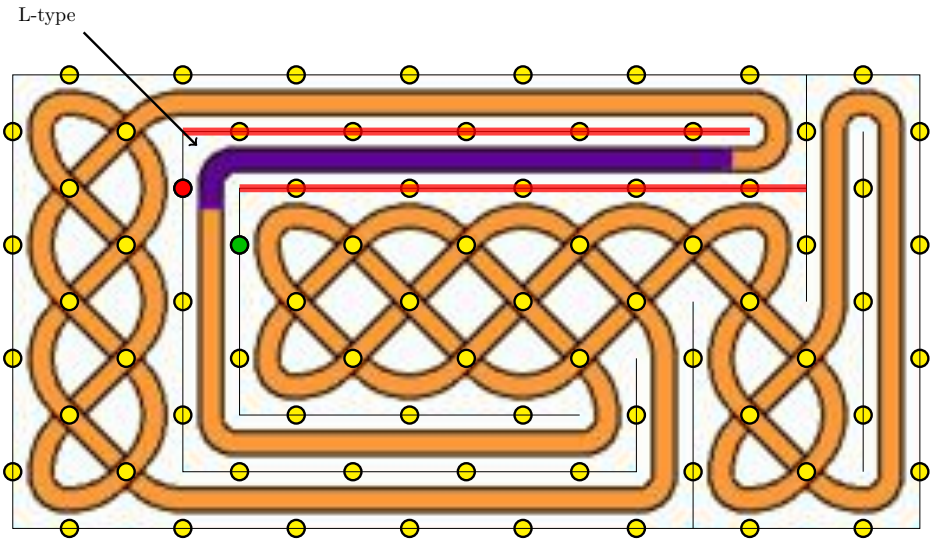


Figure 4.11: L-type corner*

There are 10 dots in the adjacent parallel lines of feature walls (5 each on the top and bottom), so we call the length of this run **10**. The first dot on the left that is *not* a part of these lines is marked in red.

(We know it's the first one off the run because if it had a horizontal feature wall, it would extend the run). It has a feature wall that is perpendicular to the original run. The dot after that is marked in green. It has a feature wall that is also perpendicular to the original run. That makes this an **L-type corner**. Note the strand makes a right angle, like the letter L.

Continuing along the component, we come to a vertical run, marked in purple below. The adjacent parallel lines of feature walls are in red.

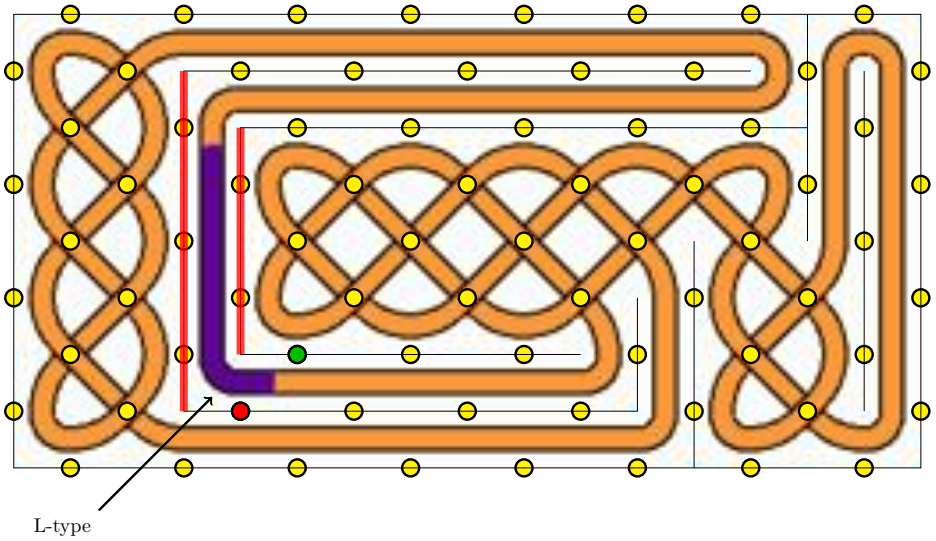


Figure 4.12: L-type corner*

The length of the run is **5**. Just like the last one, this is an **L-type** corner.

The component ends with another horizontal run. That run is in purple below, and its parallel lines of feature walls are in red.

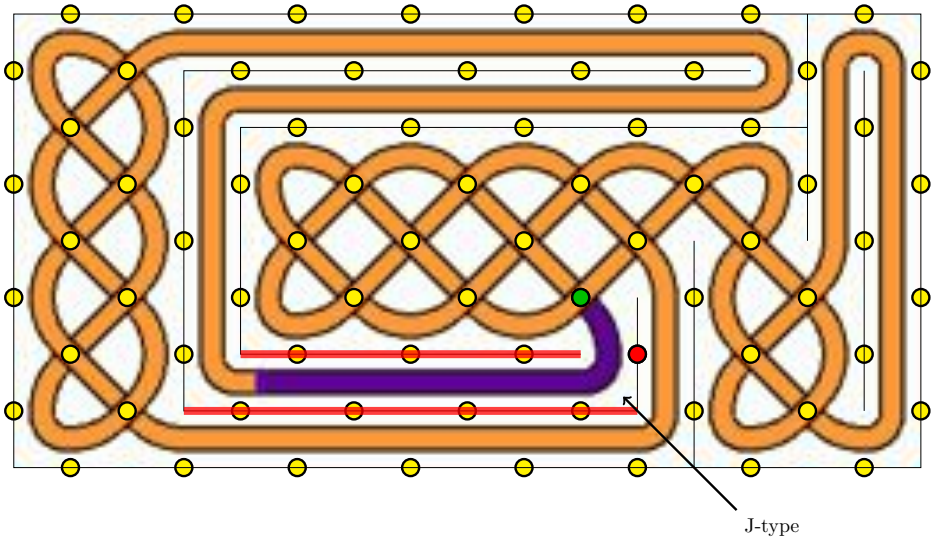


Figure 4.13: J-type corner*

The length of this run is **7**. The first dot not on the run is marked in red. (We know it's the first

one off the run because if it had a horizontal feature wall, it would extend the run). It has a feature wall perpendicular to the original run, so we move to the next dot, shown in green. This has no feature wall, so the corner is **J-type**. (Note the curve made by the strand approximates a letter J.) The component ends at the green dot.

The full component (from crossing to crossing) is shown in purple below. It can be described as follows, starting in its upper-left corner: D-type corner, run of length 11, C-type corner, run of length 10, L-type corner, run of length 5, L-type corner, run of length 7, J-type corner.

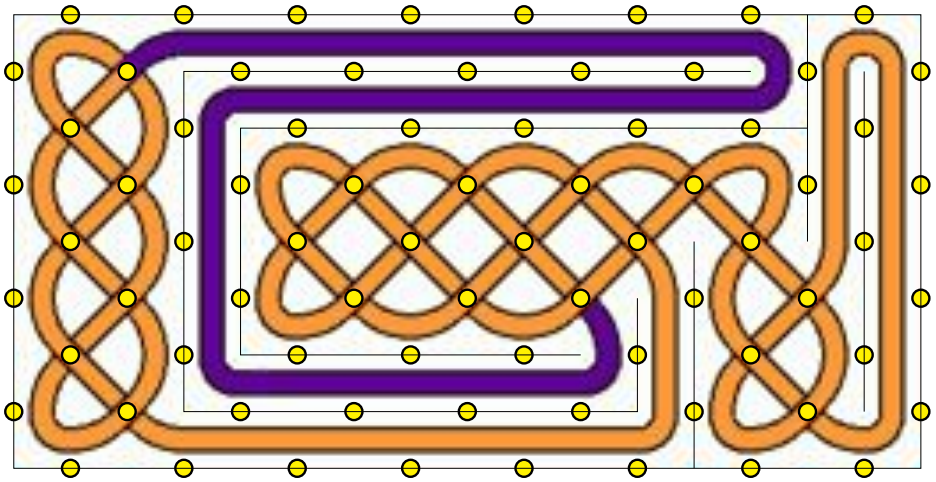


Figure 4.14: One component with four runs*

Similarly, the component shown in red below can be described (starting from the left) as D-type corner, run of length 4, C-type corner, run of length 7, J-type corner.

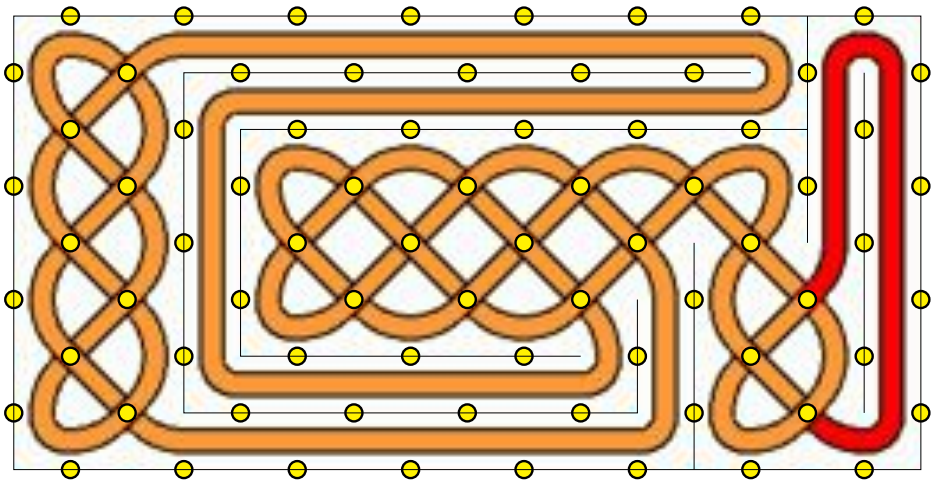


Figure 4.15: One component with two runs*

4.5 Number Intersections

In order to assemble the strands once they're done, it's helpful to number intersections.

Follow your strand around as you did when you were identifying components. Between each component, add a labelled marker with the number of the intersection, noting whether that part of the strand is on the top or the bottom of the intersection.

Often knots are symmetric in such a way that different strands can be crocheted using the same instructions. If you're planning on doing this, when you number intersections along one strand, number them along all the strands. This will probably result in several intersections having the same number. The intersection labels are helpful for assembling your final knot, but slightly ambiguous labelling is usually easy enough to deal with by carefully following your knot diagram.

4.6 Components to Patterns

Now we have a way of breaking down a knot into a sequence of components. Each component has its own pattern. So the task now is to simply match them up. If you want to make a twist, you can look up the pattern for a twist; then if you want to attach it to a quarter turn, look up the pattern for a quarter turn.

There's a lot of finagling involved: the order of the components reverses every row since we go back and forth; you need to keep track of which row you're in, and whether you're working outside-to-inside or inside-to-outside along a turn; and you need to remember which marker goes where. This is *definitely* the kind of grunt work best outsourced to a computer.

4.6.1 How to do it all by hand

If you *don't* want to use the program to assemble your list of components into a pattern, you can just look up the dimensions of each component. Stitch counts for components are given in Appendix sections A.1 (standard components) and A.2 (runs).

For example, suppose your strand has three components: straight, inner quarter, outer quarter. Let the markers between them be 1T and 2B.

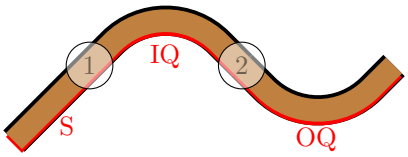


Figure 4.16: Example strand

Foundation Chain We see that each straight component has 10 stitches (Figure A.1); the innermost row of a quarter turn has 3 sts / 7 sts / 3 sts (Figure A.2); and the outermost row of a quarter turn has 3 sts / 15 sts / 3 sts (Figure A.2). So our foundation chain is:

$$10 \text{ ch } \overset{1T}{/} \text{ 3 ch / 7 ch / 3 ch } \overset{2B}{/} \text{ 3 ch / 15 ch / 3 ch}$$

Row 1 Now we're going back in the other direction, so we *start* with the outer quarter. Moving in one row from the outside, we see it has lengths 3 / 14 / 3, and that 14 is one less than 15, the previous measure (Figure A.2). Then we go to the inner quarter turn, which has measures 3 / 8 (+1) / 3 (Figure A.2). Finally, the straight section always has 10 sts (Figure A.1). Ignoring the labels, we have these directions:

3 sc / 14 sc (-1) / 3 sc / 3 sc / 8 sc (+1) / 3 sc / 10 sc

Continue like this until you've finished all rows.

4.6.2 How to avoid doing it all by hand

In an ideal world, I'd make a nifty web app that goes straight from the picture to the pattern. In the real world, even after going through a million tutorials that say it's *soooo easy*, I'm still clueless. Sorry. So what I have instead is some utilitarian Python in a Jupyter notebook. You enter the knot and your dimensions, and the program gives you a pattern.

If you're completely unfamiliar with Jupyter notebooks and Python, I've made a video tutorial here: https://youtu.be/qAq_-lJdOIg The video walks you through opening the program and entering your knot characteristics to generate a pattern. If you have used Jupyter notebooks / Python before, you probably don't need to watch the video.

I've tested the program on a number of patterns, but of course there may still be mistakes. (In particular, runs have been tested less than standard components.) All it does it automate the process of finding and concatenating arclengths, so if something looks wrong, you can spot-check by hand by comparing to this document.

How to open the notebook

If you know what a Jupyter notebook is and how to open it, you can grab it directly from the github repository here: https://github.com/ecyeager/KnotworkCrochet It's formatted to be as user-friendly as possible for people uncomfortable with programming. If you know even a little Python, you can get rid of a lot of parts. (For example, it's nicer to hard-code variables like stitch height than it is to enter them with widgets. So, you can update that variable in the appropriate code cell and then just delete its widget.) You can skip to What to do if you know some coding.

If you're not so sure about Jupyter but you still want to give things a try, click here: https://mybinder.org/v2/gh/ecyeager/KnotworkCrochet/main This uses Binder, which is a fancy way of being able to do all your code stuff in your browser, without having to install programs on your computer.

It will take a while to load. Once it's loaded, you'll see a screen that looks something like this:

Figure 4.17: Binder screen

Click on the first file, PatternGeneratorPublic.ipynb. That will load up the Jupyter notebook, which will look something like this (but longer):

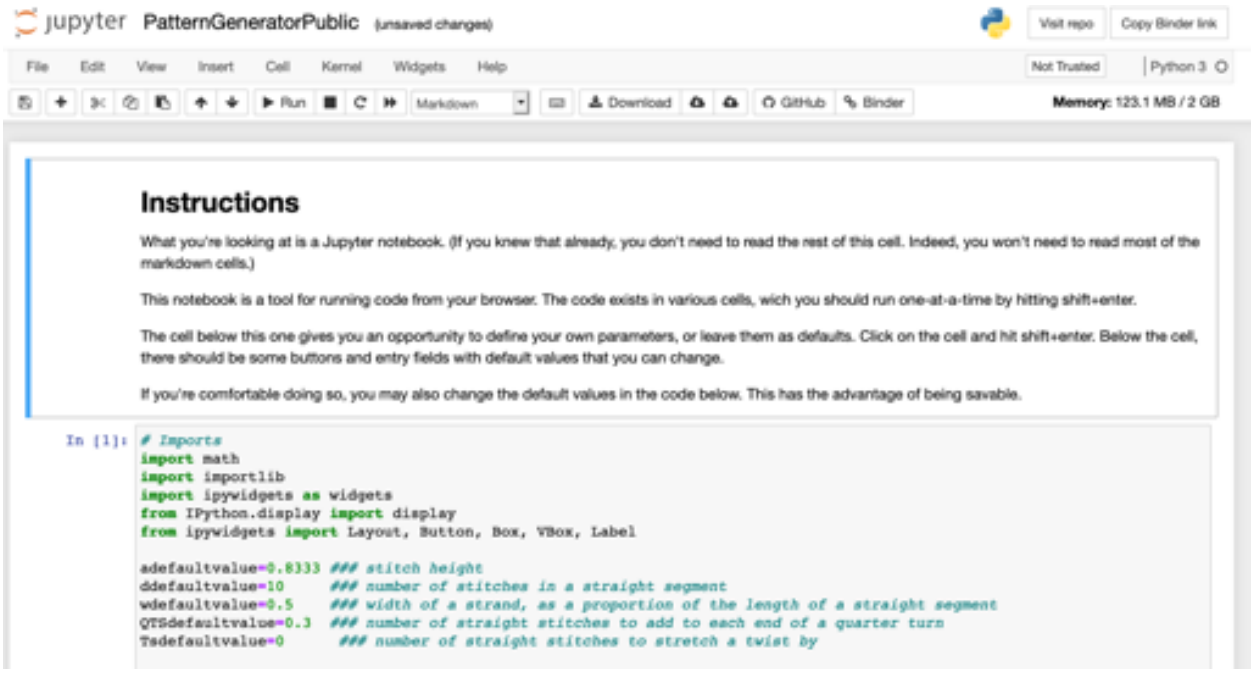


Figure 4.18: Notebook

How to use the notebook

There are alternating cells of text and code. Each code cell is run by clicking somewhere inside it and then pressing shift+enter. The text generally tells you what to do.

What to do if you know absolutely no coding

Start by clicking on the cell that starts with "#Imports" and then hitting shift+enter. You should see some buttons and text fields pop up below that cell (Figure 4.19). The defaults for your knot dimensions are pre-filled, which you can change as you like. (You might have to scroll to see all the options.)

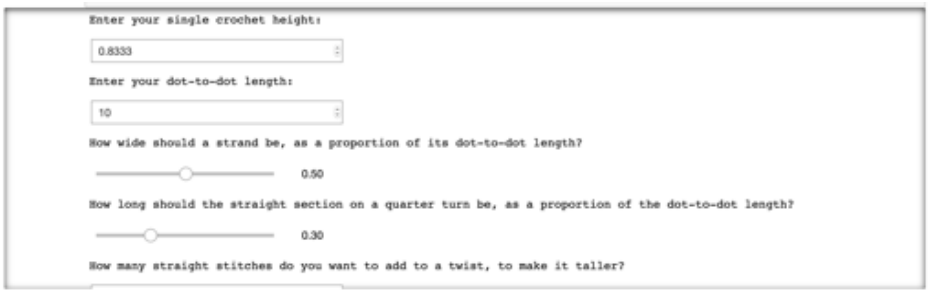


Figure 4.19: Buttons and text fields for entering dimensions

Move from each cell to the next, following the directions. Instructions should guide you through the process. Make sure you hit shift+enter for each cell with code in it.

The hardest part of the notebook is describing the knot in a way the program will understand. There are buttons to do this for you.

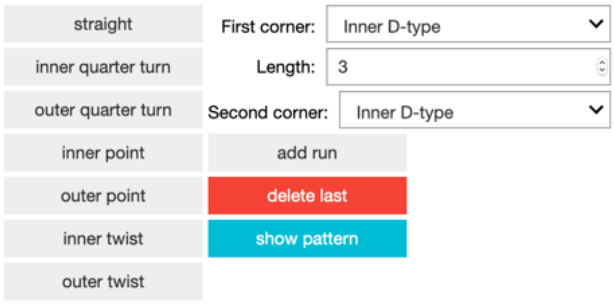


Figure 4.20: Buttons for entering the knot pattern

If your strand has components (say) "straight, straight, outer quarter turn," then click on those buttons, one-by-one, in that order. If you want to add a run, select the corner types and lengths, then click "add run." (Unfortunately I only have buttons for runs with D-type and J-type corners – not for runs that are concatenated with C-type or L-type corners.)

If you'd also like to add labelled stitch markers, then you can enter them one-by-one in the next cell. Remember to hit shift+enter to make the entry box appear.

Figure 4.21: Text box for entering the markers

One by one, enter the label you'd like for a marker, then click "add label."

At the end, when you've hit shift+enter on the final code cell, you'll see a pattern. Something like this:

```
                    print(SegmentSCOuter,stitch,"/",Tmiddle,SegmentSCInner,stitch,"/",Tmiddle,SegmentSCOuter,end)
                else:
                    print(SegmentSCOuter,stitch,changeOuter,"/",Tmiddle,SegmentSCInner,stitch,changeInner,"/",Tmiddle,S
            ###Explicitly-written strings (incl runs)
            else:
                print(seq[r],end)
print()
print("Finish off")
```

```
***Foundation Chain***
10 ch /(1T)
10 ch /(2B)
3 ch / 15 ch / 3 ch

***Row 1***
3 sc / 14 sc(-1) / 3 sc /
10 sc /
10 sc

***Row 2***
10 sc /
10 sc /
3 sc / 12 sc(-2) / 3 sc

***Row 3***
3 sc / 11 sc(-1) / 3 sc /
10 sc /
10 sc
```

Figure 4.22: Pattern generated by program

What to do if you know some coding

First, remember that we all started somewhere, and try not to get too annoyed at my lack of coding elegance. Feeling generous? OK, let's move on.

Using the widgets is harder than hard-coding the variables, not least because with the widgets you can't easily save your work. So I recommend just reading the comments and changing parameters in the code cells.

The hardest things to enter are the lists "pattern" and "markers."

pattern This is a list whose entries are components OR customized lists of instructions. You'll add them in order, as you go along your strand.

standard components If you have a standard component, enter a string: S, IQ, OQ, IP, OQ, IT, or OT (straights, inner/outer quarter turns, inner/outer points, and inner/outer twists).

custom instructions If you enter anything other than the seven strings above, it needs to be a list with one entry per row (including the foundation row). So you can have an element of the list "pattern" be a list of the form: `["instructions for row 0","instructions for row 1",...]`

If you happen to have a run, there's a separate function that will define these instructions for you.

run The function "run" takes a list as an argument. That list should have elements that alternate between corner types (ID, OD, IJ, OJ, IC, OC, IL, OL, entered as **strings**) and lengths (entered as **numbers**).

For example, suppose a run starts with an inner D-type corner, then has straight length 5, then has an inner C-type corner, then straight length 4, then an outer J-type corner. To generate the instructions for that run, you would use: `run(["ID",5,"IC",4,"OJ"])`

Example Suppose a strand has the following components:

straight, inner quarter, straight, inner D-type corner, run of length 3, inner J-type corner, straight

Then you would define this list:

```
pattern=["S","IQ","S",run(["ID",3,"IJ"]),"S"]
```

Note that the run is not entered as a string. It's a function that will *return* a string.

markers To add stitch markers to the crochet instructions, define "markers" as a list of strings. For example,

```
markers=["1T","2B","3T","4B"]
```

Remember that you have to run basically every cell. If you go back and change something at the start, you'll probably have to re-run at least some cells in order for everything to update. The final cell in the notebook will turn your pattern and parameters into crochet instructions.

Chapter 5

Translating Knots to Crochet, Part 2

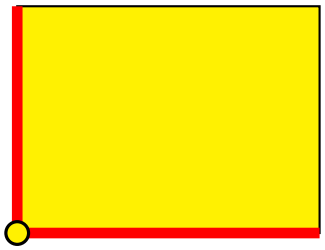
In this chapter, we detail the exact shapes assigned to each component. We only use high-school-level geometry, but we use a lot of it. If all you want to do is make crochet patterns from knots, you do not need to read this section.

5.1 Underlying Dimensions: Gauge and Strand Width

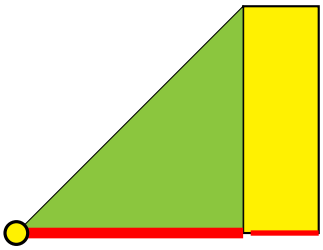
It is common in crochet patterns to assume that a single crochet is roughly square. That is, if you work a swatch of 10 sc for 10 rows, it should be as long as it is wide. For these patterns, we use slightly more precision.

We will treat the width of a sc stitch as one unit. The height is generally less than one unit. To measure your stitch height, work a swatch in sc, adding rows until it is a square.

A good size to start with is 10 sc worked for 10 rows; keep adding rows until the height is as equal as possible to the width. If you want increased accuracy, try 20 sc. If you don't have a measuring tape handy, you can check whether a rectangle is a square by folding it diagonally, explained in Figures 5.1 and 5.2.

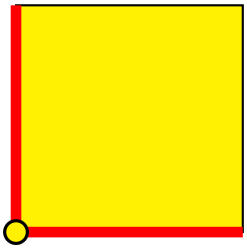


(a) Fold the swatch at the corner dot so that the red lines align.

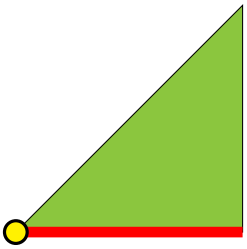


(b) The section sticking out on the right tells you this is not yet square. Work more rows.

Figure 5.1: Folding a Rectangle



(a) Fold the swatch at the corner dot so that the red lines align.



(b) When you align the two red sides, the remaining two sides line up as well. You have made yourself a square.

Figure 5.2: Folding a Square

The ratio $\frac{\text{number of sc per row}}{\text{number of rows}}$ in a square swatch is your **stitch height**. It will probably be less than 1.

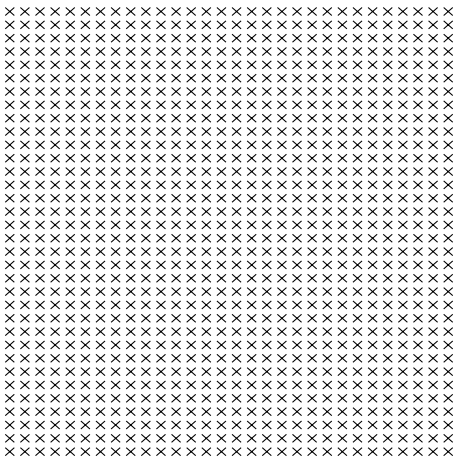


Figure 5.3: If 30 sc and 34 rows makes a square, then stitch height is $\frac{30}{34}$.

The lengths of segments in a strand are determined by the **dot-to-dot distance**. This is the distance, in stitches, between two diagonally-adjacent dots. So a straight section will have this many stitches in it.

The width of a strand works nicely as an even number: taking into account the foundation chain, even numbers make the middle of the strand easy to find.

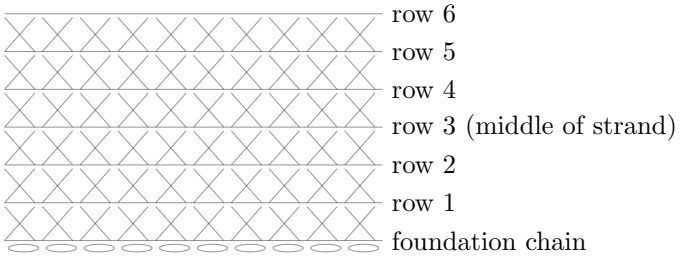
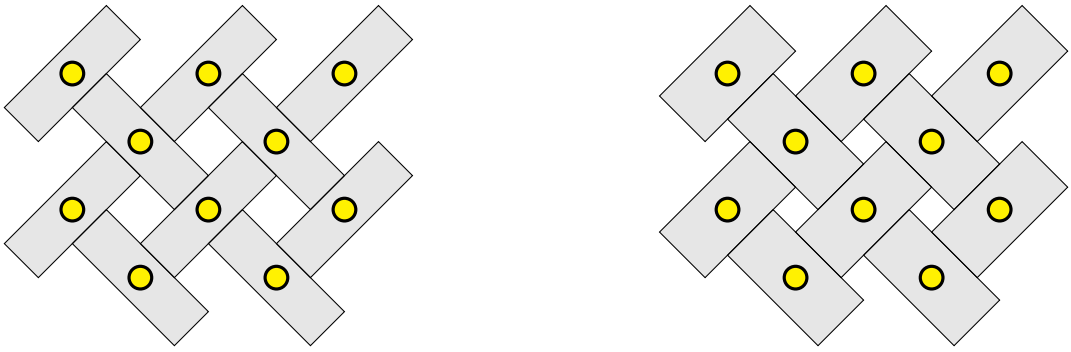


Figure 5.4: Even number of rows in a strand

If the width of a strand is one-half its dot-to-dot distance, then the spaces between strands will be as thick as the strand itself in a plain knot. For heavier items, like afghan squares, a thicker strand may make more sense. The strand width must be less than the dot-to-dot distance.



(a) Strand width one-half of the dot-to-dot distance

(b) Strand width two-thirds of the dot-to-dot distance

Figure 5.5: Strand widths

5.2 Restrictions

The general dimensions you need to decide on are:

u: This is the dot-to-dot distance, also the number of stitches in a straight segment.

s: This is the straight section on either end of the arc when you work a quarter-turn. Applies only if your knot has quarter turns. (It probably does.)

t: This is the straight section added to a twist. Only applies if your knot has a twist and you want it a little elongated.

strand width: This is the total width of one strand. It's the number of rows in the piece multiplied by your stitch height. This isn't given a variable explicitly elsewhere, but for now let's call it w. The largest value of Δ (used later on as the distance away from the centre of the strand) is $\frac{w}{2}$.

In order to keep your knot from unintentionally overlapping itself, these dimensions have certain restrictions. For each component, I calculated the relationships between dimensions that will keep the component from crossing its feature wall(s). These are given in Table 5.1.

If a component does not appear in your knot, then it is possible to ignore the restriction. Since your actual knot is unlikely to match the planned dimensions exactly, it's nice to build in some error tolerance. If you violate a restriction but you really want to keep the dimensions as they are, you can probably fix the relevant parts of your pattern ad-hoc.

component(s)	restriction	notes
straight	$w \le u$	You'll want to keep away from $w = u$, unless you specifically want your strands to bunch up.
quarter turn	$s \le u - (\sqrt{2}+1)\frac{w}{2}$	The value of s has a huge impact on how your quarter turns look. Bigger s-values make a more dramatic arc.
point	$w \le u\left(2-\sqrt{2}\right) \cdot$ (stitch height)	If we take the number of rows as $\frac{w}{\text{stitch height}}$, this is equivalent to (number of rows in a strand) $\le u\left(2-\sqrt{2}\right)$
twist	$w \le u\left(2\sqrt{2}-2\right)$ and $t \le \left(\sqrt{2}-1\right)u - \frac{w}{2}$	Remember this causes a dangly piece, so consider the final use of your object before you make a knot with a twist component.
J-type and C-type corners	$w \le \frac{u}{\sqrt{2}}$	Have I mentioned yet that runs are a pain in the neck? If your J-type or C-type corner is at the outside edge of the piece, then there's nothing for it to run into, so this is somewhat easier to ignore. You can also reduce the length of the attached runs to pull a corner away from an overrun feature wall.

Table 5.1: Restrictions on dimension parameters

5.3 Side Length

Note that the distance between two dots that are *horizontally* (or vertically) adjacent is the square root of 2 times the dot-to-dot distance.

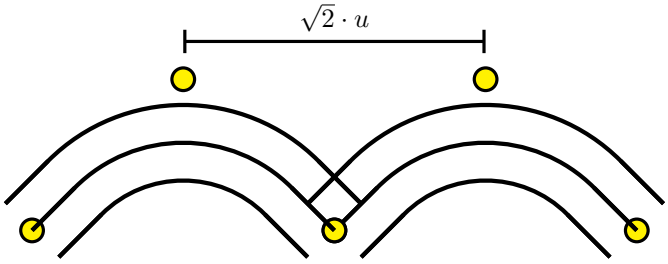


Figure 5.6: Border Measure

So if your dot-to-dot distance is, say, 10 stitches, then the number of stitches between peaks on your border will be about 14, since $10\sqrt{2} \approx 14$.

This is useful for planning the final dimensions of a piece, or for adding a straight border.

5.4 Yardage

Consider a blank knot on a grid of $n \times m$ crossings (highlighted dots), or $(2n+1) \times (2m+1)$ dots in the underlying grid. (The blank knot in Figure 3.2 has $n = m = 5$.) The combined length of all strands in the blank knot is $4nm$ times the dot-to-dot length. So as a very rough estimation, the yardage needed on a knot of this size will be around $4nm$ times the amount needed on a swatch that is the size of one straight segment.

You'll need to know how much yarn you use when you work a swatch of one straight segment. Having a small scale is easiest, but it's not the only way. If you don't have a scale, you can work the swatch, then

unravel and measure it. You can also mark every metre (or every two metres) of yarn before you use it, avoiding the need to disassemble the swatch.

For example, suppose you're working a 7-by-7 knot. Your total yardage should be about $4 \times 7 \times 7 \approx 200$ times the amount used in a straight segment. If your straight segment uses 1 m of yarn, your total yardage will probably be around 200 m. If your straight segment uses 2 m of yarn, your total yardage will probably be around 400 m, and so on.

The Python program does include a yardage calculator at the end, but yardage requirements are always only estimates. Also note that the calculator gives yardage for one copy of the strand whose pattern you supplied. If you need to make multiple strands, add their requirements together.

5.5 Component Geometry

Intersections occur at highlighted dots, where the strands should meet at right-angles to one another. So given our usual orientation, we can assume every segment starts and ends at a dot, with initial and final angles 45 or -45 degrees.

In general we will denote by u (for "unit") the dot-to-dot length, and we assume the stitch width is 1. That way, length and arclength are measured in stitches.

Each row has height equal to the stitch height; we will use the shorthand Δ for the amount added or subtracted from the middle row. So for the middle row, $\Delta = 0$; for the row before resp. after the middle row, $\Delta = -$(stitch height) resp. $\Delta = +$(stitch height); etc.

Segments are worked as straight pieces, right angles, and circular arcs. For curves, we use circular arcs so that we can work our increases uniformly throughout the curve, and easily compute arclength.

5.5.1 Straight

All rows have the same length on a straight segment: the dot-to-dot length, u.

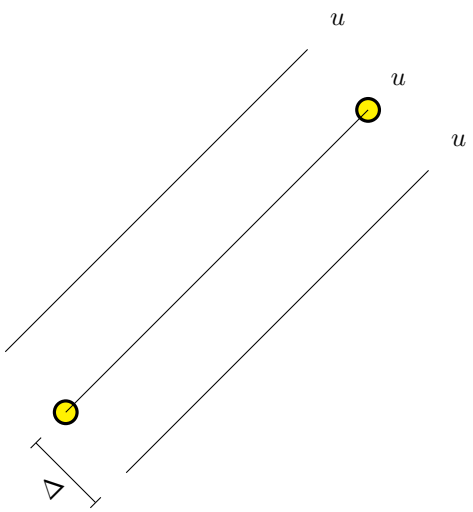


Figure 5.7: Straight segment

5.5.2 Quarter Turn

Using a quarter-circle for a quarter-turn in a knot results in a fairly flat, uninspiring curve. To make the curve more pronounced, the segment begins and ends with a short straight section.

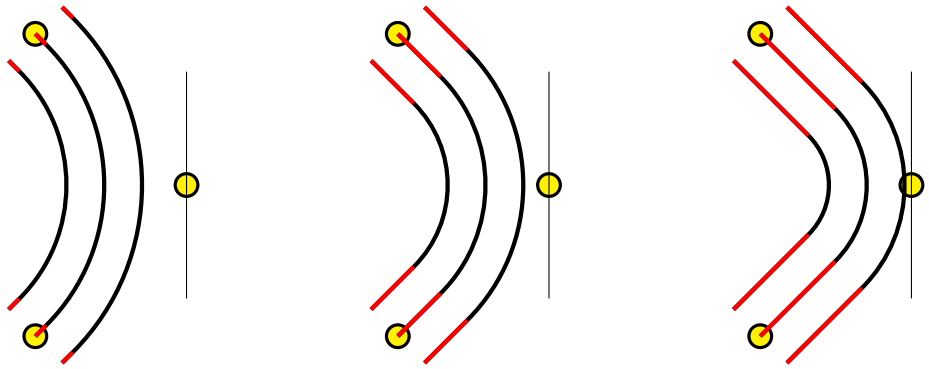


Figure 5.8: Quarter turns with straight segments of varying lengths shown in red

Let the length of the straight segment be s. This is the same for all rows. The radius of the quarter-turn is $u - s + \Delta$, so the arclength of the quarter-circle segment is $\frac{\pi}{2}(u - s + \Delta)$.

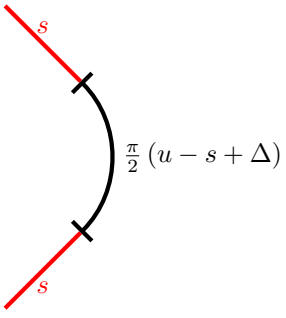


Figure 5.9: Quarter turn lengths

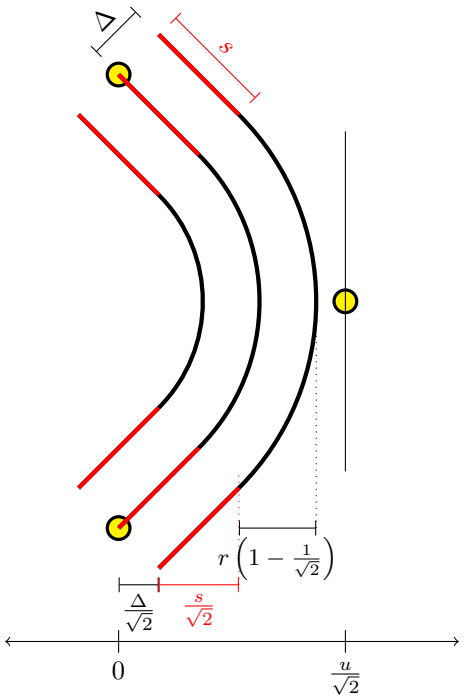


Figure 5.10: Outer horizontal lengths in quarter turns

As it is drawn above, the strand moves a horizontal distance of $\frac{s}{\sqrt{2}}$ in the red straight section, and an additional horizontal distance of $r\left(1-\frac{1}{\sqrt{2}}\right)$ in a quarter-turn of radius r. Suppose the outermost edge of the strand lies a distance of Δ away from the strand's centre. (So $\Delta = \frac{\text{total number of rows}}{2} * (\text{stitch height})$.) This outermost edge starts the component a horizontal distance of $\frac{\Delta}{\sqrt{2}}$ to the right of the dot. To avoid crossing over the feature wall (and running into a different strand where there should not be an intersection), we need:

$$\frac{u}{\sqrt{2}} \geq \frac{\Delta}{\sqrt{2}} + \frac{s}{\sqrt{2}} + r\left(1-\frac{1}{\sqrt{2}}\right)$$

Since $r = (u - s + \Delta)$,

$$u \geq \Delta + s + (u - s + \Delta)\left(\sqrt{2}-1\right)$$

So,

$$s \leq u - (\sqrt{2}+1)\Delta$$

For quarter-turns on the outside border of the knot, there's nothing to run into, so crossing the feature wall is perhaps acceptable. Otherwise, a large value of s risks the quarter turn overlapping with a different part of the knot, in a place where no intersection is intended.

5.5.3 Point

A point is the only standard component with a sharp corner. We work it as two eighth circles that meet at a right angle. The eighth circles have radius $u + \Delta$.

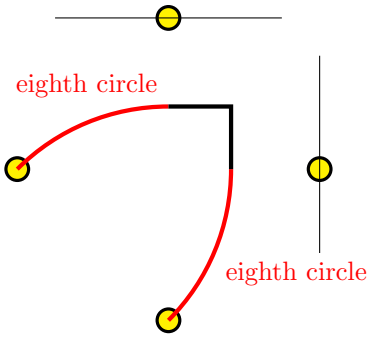


Figure 5.11: Point

We might have liked the straight segments to have a length that changes by one stitch height h each round, but in practice this is awkward. Instead, our change per round is one stitch *width*. (That comes from simply adding 2 ch in each row to make the corner increases.) Given a value $\Delta = h*$(row number), the quantity (stitch width)*(row number) is the same as (row number), or $\frac{\Delta}{h}$.

When $\Delta = 0$, we would like the straight segments to have length $u\left(1 - \frac{1}{\sqrt{2}}\right)$. However, there's another consideration: the innermost row shouldn't have a negative measure for the straight part. So the straight part measure, S, should be at least half your total number of rows. (If the inner most row has measure at least one, that's even easier, since you won't have to create a new component in the first row.) Therefore, let $S = \max\left\{u\left(1 - \frac{1}{\sqrt{2}}\right), \left\lceil \frac{\text{total number of rows}}{2} \right\rceil\right\}$.

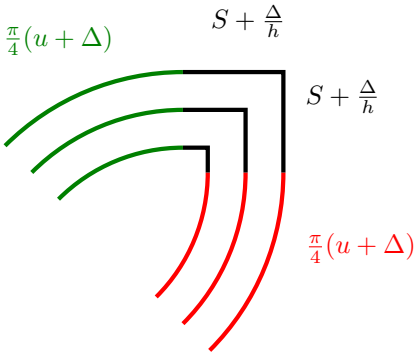


Figure 5.12: Point lengths

The length of one side of the straight section, in the middle of the strand, should be S. If we are starting at the inside of the point and increasing, that means the middle row of the point is worked as: $(S-1)$ sc, 2 ch, $(S-1)$ sc. If we are starting at the outside of the point and decreasing, that means the point is worked as $(S-1)$ sc, 2 dec, $(S-1)$ sc. Or, to make a sharper point: S sc, skip 2 sts, S sc.

The straight-part length changes by 1 each row. So if your row is x rows from the middle (where x is negative for inner rows and positive for outer rows), then the length of each straight part of the point is $S + x$. In the foundation chain, you simply work $2(S + x)$ chain stitches. For the other rows, you work $(S + x - 1)$ sc, 2 ch, $(S + x - 1)$ sc if you are going from the inside of the point to the outside, and you work $(S + x - 1)$ sc, 2 dec, $(S + x - 1)$ sc OR $(S + x)$ sc, skip 2 sts, $(S + x)$ sc if you are going from the outside of the point to the inside.

5.5.4 Twist

A twist in total turns the strand three-quarters of a circle, but simply working three-quarters of a circle results in a nubby-looking component. An elongated component looks more harmonious. So, a twist consists of two eighth-circles of radius $u + \Delta$ connected by a half-circle of radius $u\left(1 - \frac{1}{\sqrt{2}}\right) + \Delta$.

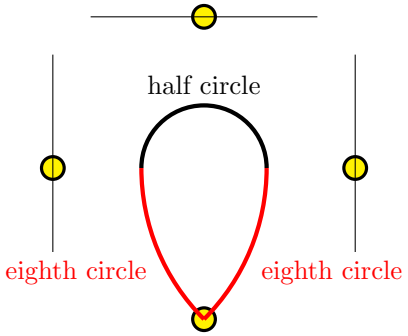


Figure 5.13: Twist made up of three arcs

So, the eighth-circles each have arclength $\frac{\pi}{4}(u + \Delta)$, and the half-circle has arclength $\pi\left(u\left(1 - \frac{1}{\sqrt{2}}\right) + \Delta\right)$.

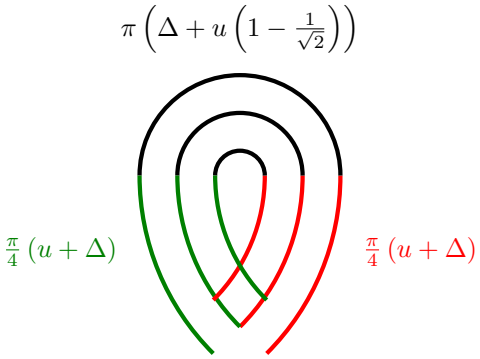


Figure 5.14: Twist segment lengths

Be aware that if the ratio between consecutive segment lengths is greater than two (or less than one-half), working (2 sc in one stitch) for increases or slst2tog for decreases won't be enough. Alternatives are adding chains between stitches / skipping stitches, or working (n sc in one stitch) / slst n tog.

If you want the twist to be longer, you can insert straight segments on either end of the half-circle segment. These will have the same measure every row.

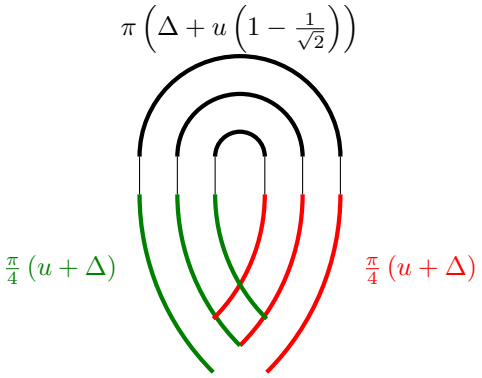


Figure 5.15: Elongated twist

Let Δ be the largest distance from the centre of the strand to its edge. The radius of the eighth-turn is $r_1 = u + \Delta$, and the radius of the half-turn is $r_2 = \Delta + u\left(1 - \frac{1}{\sqrt{2}}\right)$. From the dot marked in green, the eighth-turn has a vertical height of $\frac{r_1}{\sqrt{2}}$, and the half-turn has a vertical height above that of r_2. The outside edge of the strand starts a vertical distance of $\frac{\Delta}{\sqrt{2}}$ *below* the green dot.

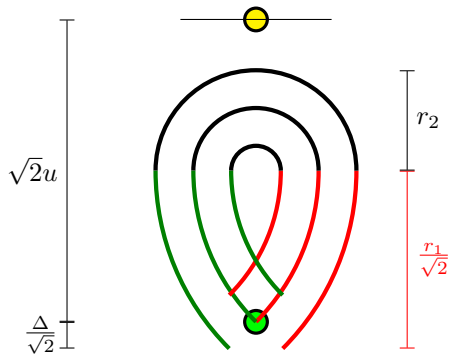


Figure 5.16: Vertical distances of a twist

To avoid overrunning the feature wall, we must have:

$$\begin{aligned} \frac{r_1}{\sqrt{2}} + r_2 &\leq \frac{\Delta}{\sqrt{2}} + \sqrt{2}u \\ \frac{u + \Delta}{\sqrt{2}} + \left(\Delta + u\left(1 - \frac{1}{\sqrt{2}}\right)\right) &\leq \frac{\Delta}{\sqrt{2}} + \sqrt{2}u \\ \Delta &\leq \left(\sqrt{2} - 1\right) u \end{aligned}$$

If you want to add a straight section of length t to either end of the half circle, then also you'll need

$$t \leq \left(\sqrt{2} - 1\right) u - \Delta$$

5.6 Runs

5.6.1 General Anatomy of a Run

A **run** is formed by adjacent rows of feature walls. In order to efficiently describe runs, we separate them into a middle straight segment and the ends of those segments (corners).

The length of a middle straight segment is based on the (combined) number of feature walls in the two adjacent rows. We position the strand directly between the two adjacent rows of walls.

There are four types of corners, which we'll call D, J, L, and C. An end of a run may bring the strand to a crossing, ending the segment, or it may attach another run to the segment. The four types are described below.

D(irect) A direct (D-type) corner happens when the first dot after the adjacent rows has no feature wall. This ends the segment.

J A J-type corner happens when the **first** dot after the adjacent rows has a feature wall; then the **next** dot after that has no wall. This ends the segment.

L An L-type corner happens when the **first** dot after the adjacent rows has a feature wall; then the **next** dot after that has a wall that is perpendicular to the original run. This attaches a new run to the segment, perpendicular to the last run.

C A C-type corner happens when the **first** dot after the adjacent rows has a feature wall; then the **next** dot after that has a wall that is parallel to the original run. This attaches a new run to the segment, parallel to the last run.

5.6.2 Straight Middle Part

In order to leave room for all four types of ends, we measure the straight middle part as in Figure 5.17 below.

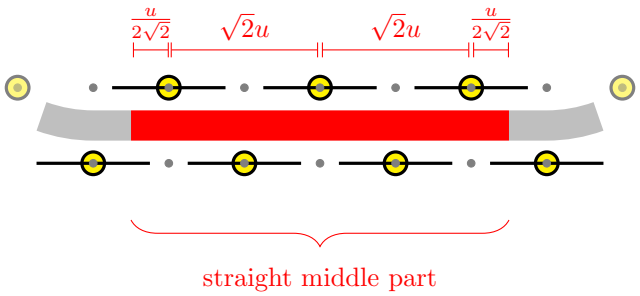


Figure 5.17: Straight middle part of a generic run

So, if there are N feature walls in adjacent parallel lines, the length of the straight part is

$$\left(\frac{N-2}{\sqrt{2}}\right)u$$

5.6.3 D(irect)-Type Corner

A D(irect)-type corner should turn a total turn of $\frac{\pi}{4}$ radians. Then it will meet its dot (which is the barrier between two components, and the site of a crossing) with the correct diagonal orientation. So, we work a D-type corner as an eighth-turn.

We choose a radius that will position the strand in the middle of the two rows of walls at the end of its eighth-turn. Setting the radius of the eighth-turn to $\frac{u}{2\sqrt{2}-2} = u\left(\frac{\sqrt{2}+1}{2}\right)$ for the centre of the strand means that the height above the starting position will be $\frac{u}{2\sqrt{2}}$: half-way between the two lines of feature walls. At the end of the arc, the strand has travelled $\frac{u}{2\sqrt{2}}$ in the vertical direction and $\frac{u}{4-2\sqrt{2}} = u\left(\frac{\sqrt{2}+1}{2\sqrt{2}}\right)$ in the horizontal direction.

The strand has travelled $\frac{u}{4-2\sqrt{2}}$ in the horizontal direction. In order to reach the middle segment, we need to travel a total of $\frac{3}{2\sqrt{2}}u$ in the horizontal direction. So to finish this end, we need to add a straight segment of length $\left(\frac{\sqrt{2}-1}{2}\right)u$.

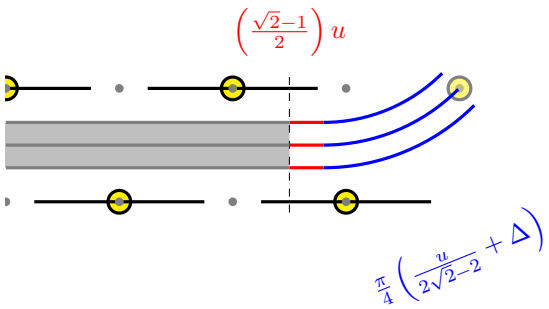


Figure 5.18: D-type corner Lengths

5.6.4 J-Type Corner

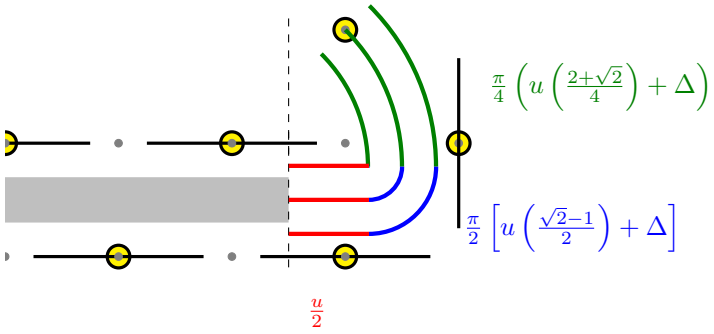


Figure 5.19: J-type corner Lengths

From the run:

- Straight a distance of $\frac{u}{2}$ (all rows)
- Quarter-turn with radius of $u\left(\frac{\sqrt{2}-1}{2}\right)$
- Eighth-turn with radius of $u\left(\frac{2+\sqrt{2}}{4}\right)$

Let Δ be the largest distance from the centre of the strand to its edge.

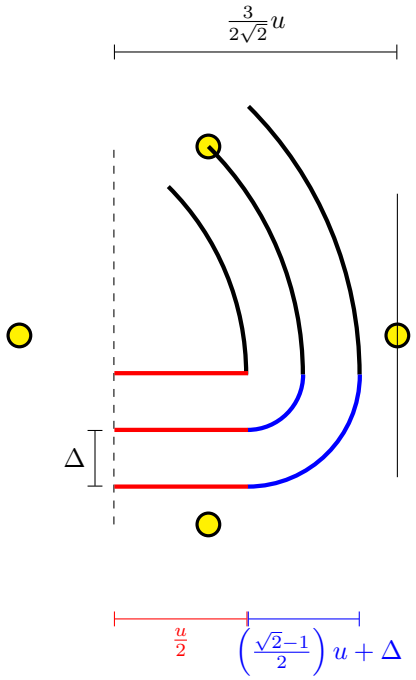


Figure 5.20: Horizontal distances in J-type corner

From the dashed line, the strand moves a straight horizontal distance of $\frac{u}{2}$; then makes a quarter turn of radius $\left(\frac{\sqrt{2}-1}{2}\right)u+\Delta$. The quarter turn moves the strand a further distance of $\left(\frac{\sqrt{2}-1}{2}\right)u+\Delta$ horizontally. In order to not overrun the feature wall, we need:

$$\frac{u}{2}+\left(\frac{\sqrt{2}-1}{2}\right)u+\Delta \leq \frac{3}{2\sqrt{2}}u$$
$$\Delta \leq \frac{u}{2\sqrt{2}}$$

For the inner edge, it might be the case that the radius for the quarter-turn might be negative. As a quick fix, I just take the negative arclength from the quarter turn and add it to the arclength for the eighth-turn, leaving the quarter turn as a segment with 0 stitches for the innermost rows. It's imprecise, but for small discrepancies seems to work okay.

5.6.5 L-Type Corner

An L-type corner is simply a right angle. It connects two runs together, and so it only appears in the middle (not the end) of a component.

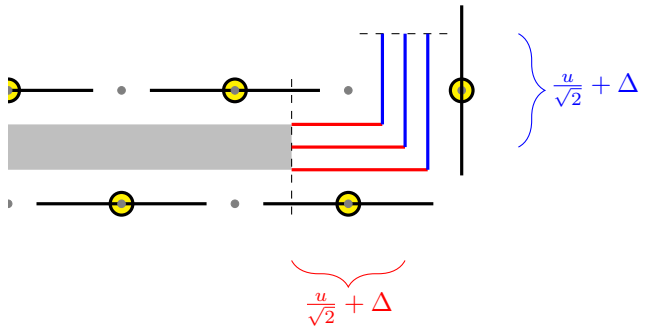


Figure 5.21: L-type corner Lengths

Make the corners by using 2 chains (working inner to outer) or 2 decrease stitches (working outer to inner). The length of each segment in the middle of the strand (when $\Delta = 0$) is $\frac{u}{\sqrt{2}}$. As with the point, for practicality, increase/decrease one each row, rather than trying to match the exact lengths.

Note: Chapter 4 uses a different convention for which stitches are part of the middle bit, and which are part of the corner.

5.6.6 C-Type Corner

A C-type corner is a half circle of radius $\frac{u}{2\sqrt{2}}$. It connects two runs together, and so it only appears in the middle (not the end) of a component.

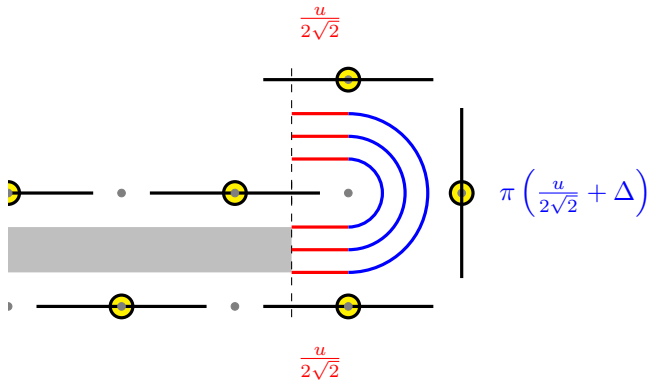


Figure 5.22: C-type corner Lengths

To centre the half-circle, we add a straight segment of length $\frac{u}{2\sqrt{2}}$ on either end. This measure is the same for all rows.

Let Δ be the largest distance from the centre of the strand to its edge. At the outermost edge, the horizontal distance moved by the half-circle is its radius, $\frac{u}{2\sqrt{2}} + \Delta$. In order to avoid running into the feature walls, we should have

$$\begin{aligned} \frac{u}{2\sqrt{2}} + \Delta &\leq \frac{u}{\sqrt{2}} \\ \Delta &\leq \frac{u}{2\sqrt{2}} \end{aligned}$$

This is the same bound we found for J-type corners in 5.6.4.

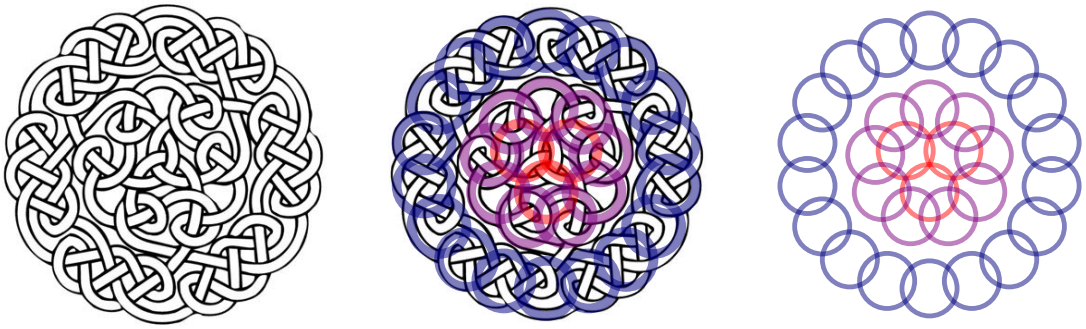
The half-circle has a vertical span twice its horizontal span, and twice as much space to move in before it hits the feature walls. So it suffices to consider the horizontal bounds.

Chapter 6

Variations and Future Directions

6.1 Non-rectangular knot constructions

The types of knots described in this document are all based on rectangular grids. Other grids are possible, and would certainly lend themselves to a similar approach. Circular knots would be particularly interesting, as the position of a component in a knot (in particular, its distance from the centre) would come into play.



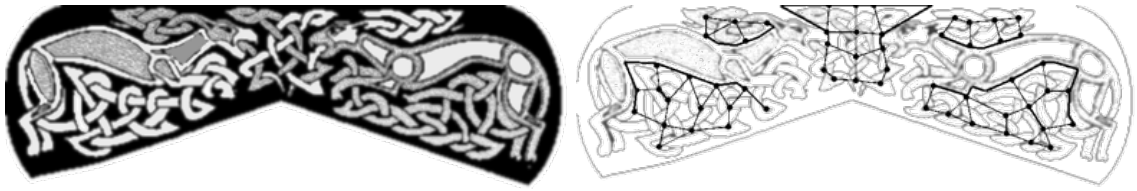
(a) Round knot (b) Overlay of knot and grid (c) Circular grid

Figure 6.1: Round knot with grid[1]

A much more general method of drawing knots can be found at http://www.entrelacs.net[2]. This method has basically no limitations on where intersections can be placed, foregoing the regularity we've made heavy use of here. Nonetheless, I believe a similar method would work to algorithmically generate crochet patterns for these knots. Using the graph paradigm of the method of drawing described in that source, one would have to take into account the lengths of the edges and their angles of incidence.

[1]Knot icon from mandalakleurplaten.nl, http://www.mandalakleurplaten.nl/kleurplaten/keltischemandalas/wirwar.html Accessed May 2021

[2]Mercat, Christian and Abbott, Steven. Celtic Knotwork, the Ultimate Tutorial, http://www.entrelacs.net Accessed May 2021



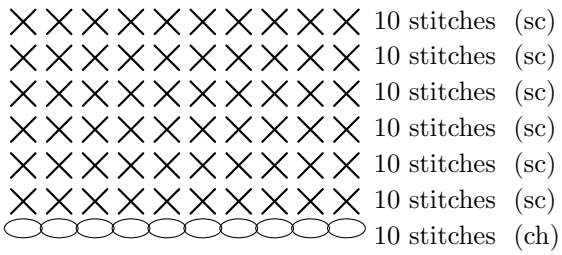
(a) A free-form knot worked around animal shapes (b) Graphs underlying the knotted sections

Figure 6.2: Free-form knot with graph[3]

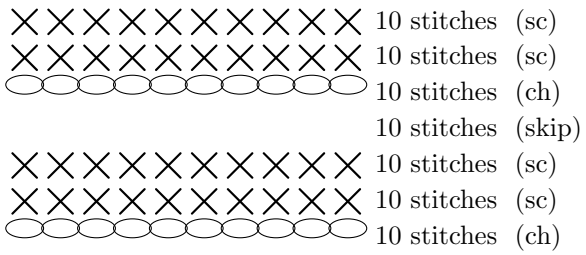
6.2 Limitations of existing program

The Python program was written with several simplifying assumptions. Notably, it forces an even number of rows, and assumes every row has the same types of stitches. The general method described here will work for any number of rows, and varying stitch heights. It's not particularly hard to make the Python accept those as well.[4]

Interlacing strands (using double and triple strands in parallel) follow the same principles as single strands. If you tweak the Python to allow for different stitch heights, you can run the code as if you are working one strand, but some of the rows are empty space and more than one are foundation chains. Alternately, using the program as written with the given defaults, a six-row strand can be worked as two parallel strands by skipping Row 3 and using Row 4 as a new foundation chain.



(a) Charted straight section stitch counts from Figure A.1



(b) Straight section charted as two parallel strands

Figure 6.3: Parallel strands

Shaping stitches could be planned for a component with all of its rows considered, rather than shaping each row ad-hoc. It's possible to think of a cluster (like in wattle stitch) as one unit, treating it like a single crochet, but then you have quite large stitches, which cuts down on your options for placing increases and decreases. For patterns that rely on modular repetition (e.g. chain every third stitch), increases and decreases would have to be more carefully thought out in order to maintain the correct number of stitches, and the correct alignment of adjacent rows. For both these cases, it would be nice to have explicit shaping instructions, rather than just "(+1)".

The program has a yardage calculator that uses some simplifying assumptions. I didn't worry too much about exactness here because of the inherent inaccuracies in calculating yardage requirements. For example, it counts stitches without differentiating between foundation chains, sc, inc, and dec stitches. The yardage is based on a straight-section swatch, and I am assuming that the ratio of chains to stitches is roughly constant

[3]Mercat, Christian and Abbott, Steven. Celtic Knotwork, the Ultimate Tutorial, http://www.entrelacs.net/IMG/gif/2139.gif and http://www.entrelacs.net/IMG/gif/2141.gif Accessed May 2021

[4]The version on the public repository is simplified for easier use, as well as greater robustness – I didn't trust myself to sufficiently error-test the versions with the noted modifications.

in all components. Furthermore, runs use global variables, meaning that running things more than once results in inflated counts. All these could be improved.

6.3 Construction techniques

Crocheting in the round would give a seam-free finish, but the initial weaving of the foundation chains would be quite tricky. Developing an easy-to-implement method for this would be lovely.

Sewing the piece together is the best assembly method I've come up with so far[5], but I still find it unpleasant. Improvements would be welcome here as well.

[5] I've experimented with adding outlines *after* the piece is woven together, slip-stitching the outline through both layers at crossings to secure them. If you look at pictures of Ravelry, you might find some pieces worked in that manner. In addition to being awkward to implement, that method also spoils the knot on the back of the work. However, it does fasten the pieces very securely.

Chapter 7

Acknowledgements and Fair Use

Knot pictures in figures marked with * , and also the diagrams used in graphics on either side of the table of contents, were generated by https://w-shadow.com/celtic-knots/. Other included images are cited where used.

The purpose of including all the back-end details is to allow you to use this method to make your own patterns! If you publish a pattern using this method, include a citation. You can cite this work as: Yeager, Elyse A General Method for Knotwork Crochet (2021) https://github.com/ecyeager/KnotworkCrochet/

Appendix A

Stitch Counts

A.1 Stitch counts for Standard Components

For each component, you just need to note how many stitches there are in each row. If you don't want to use the Python code, these are given in the figures that follow. We set the dot-to-dot distance to 10 stitches, the stitch height to 0.8696[1], and the number of rows to 6. For quarter turns, we include 3 stitches on either side worked straight. For twists, there is no straight section between the arcs.

In a **straight** segment, work 10 stitches on every row.

Figure A.1: Straight Section Stitch Counts

A **quarter turn** has three sections (so two internal, unlabeled stitch markers). Start with 3 sc; then make an arc; then 3 sc to end.

[1] based on a stitch height of 60/69; the default in the program is actually 5/6

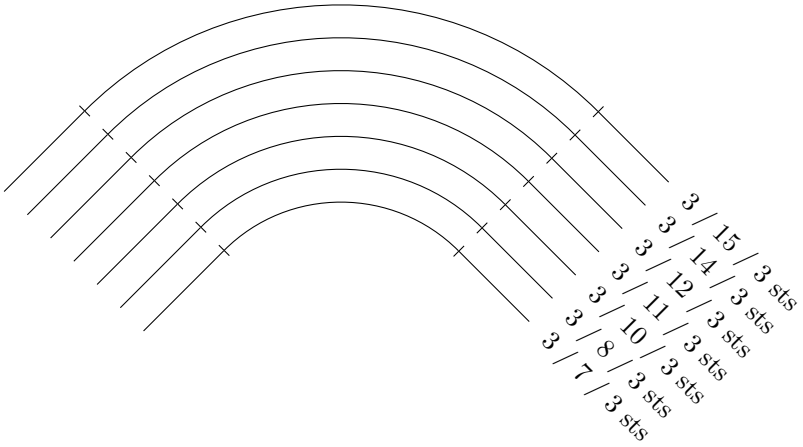


Figure A.2: Quarter Turn Stitch Counts

A **point** has two arcs on either end, and a right angle in the middle. The point of the right angle is formed by chains (working from the inner edge) or decreases (working from the outer edge). Two (internal, unlabelled) stitch markers separate the arc sections from the point section.

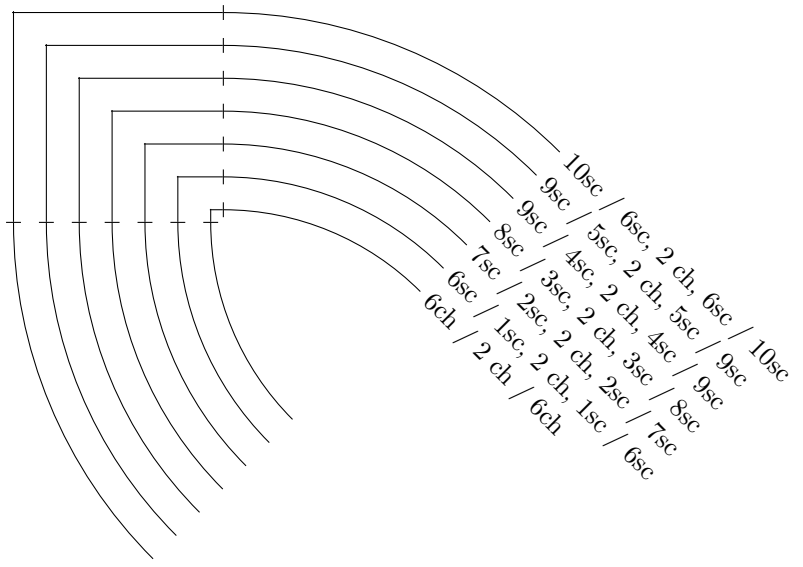


Figure A.3: Point Stitch Counts, Starting at Inner Edge

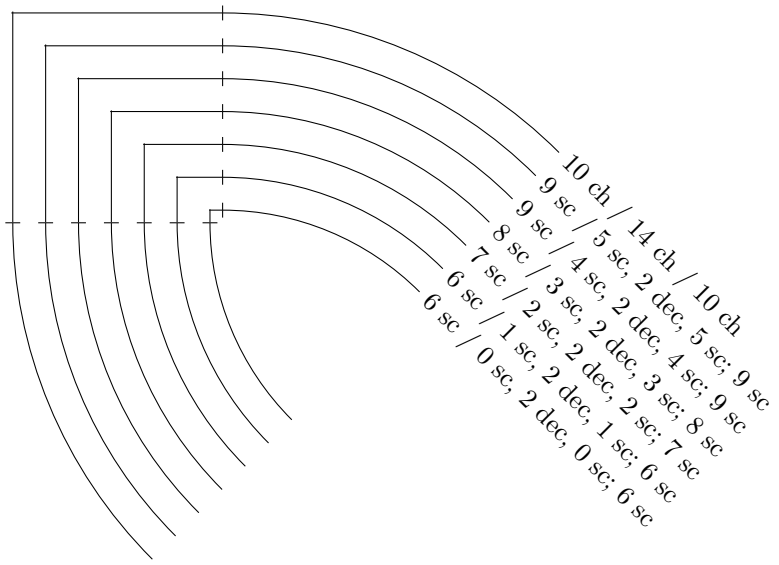


Figure A.4: Point Stitch Counts, Starting at Outer Edge

A **twist** has three sections, all arcs. So there are two (internal, unlabelled) stitch markers.

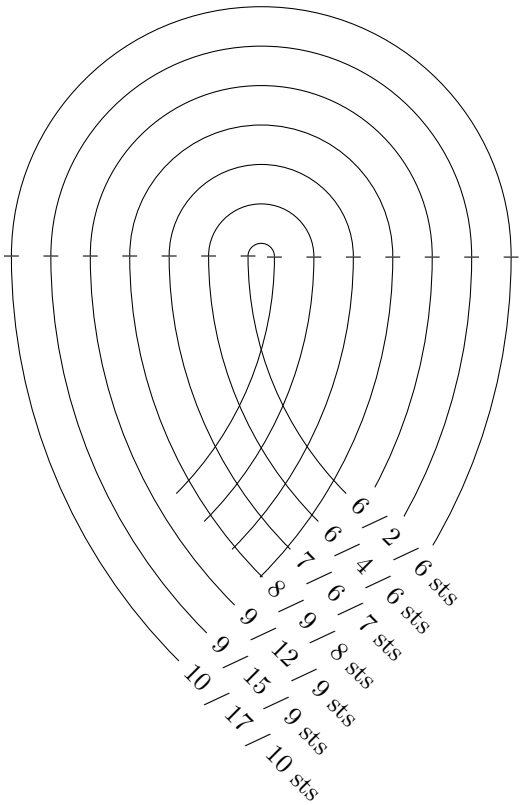


Figure A.5: Twist

A.2 Stitch counts for Runs

Each corner type has its own instructions, given in Figures A.6, A.7, A.9, and A.10. The runs between two corners are worked as straight segments, with the same number of stitches in each row. The number of stitches in these segments depends both on the length of the run (how many dots are involved in the parallel adjacent feature walls) and on the types of corners at either end of the run. Tables A.1 through A.12 give these measurements for runs of length 2 through 13.

Figure A.6 gives the number of stitches in each row of a D-type corner. Put a stitch marker at either end of this corner. The red dotted lines indicate where the straight section of the run will go.

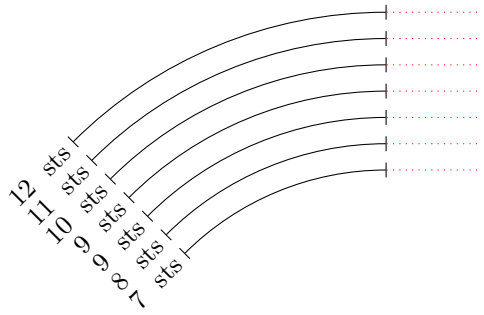


Figure A.6: D-type corner stitch counts

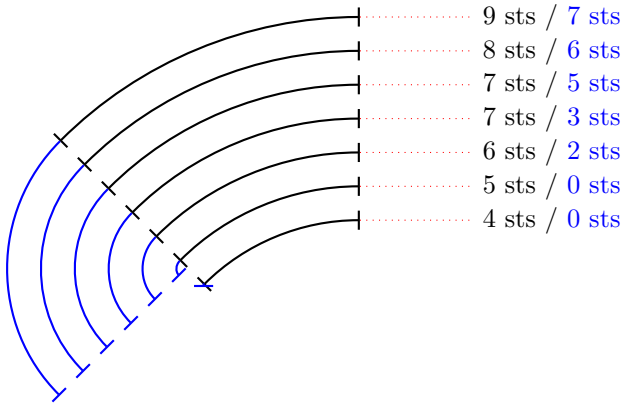
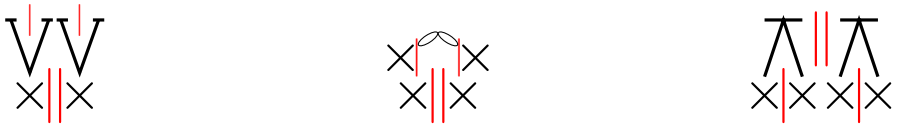


Figure A.7: J-type corner stitch counts

Note for the innermost two rows, one segment has *no stitches*. When increasing, you'll have to make that segment out of nothing in Row 2. You can either add increases to the stitches before and after the segment, or work chain stitches. When decreasing, you'll have to go from 2 stitches to 0 stitches in Row 5. You can do this by working two decrease stitches across the segment itself and the stitches adjacent to it.

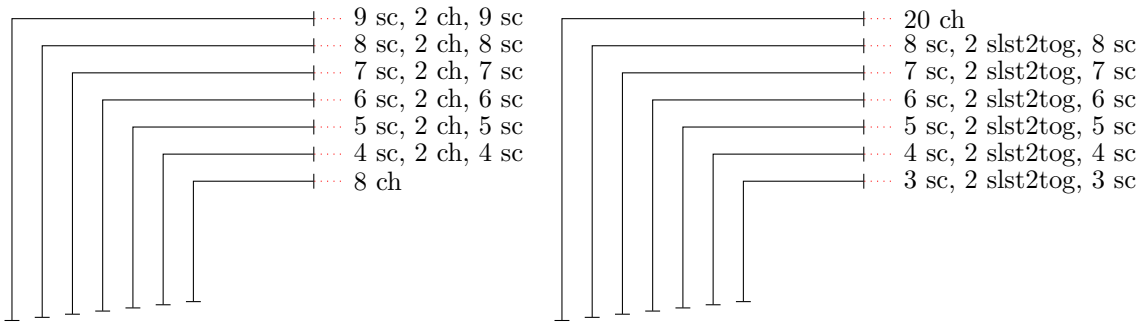


(a) Increase: make new segment using increase stitches

(b) Increase: make new segment using chains

(c) Decrease using decrease stitches

Figure A.8: Schematic for starting/ending segments with 0 stitches. Running stitch markers shown in red.



(a) inner (b) outer

Figure A.9: L-type corner instructions

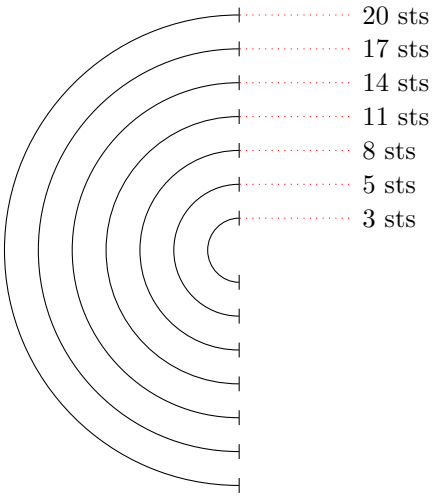


Figure A.10: C-type corner instructions

Length: 2	D	J	L	C
D	4	7	2	6
J	7	10	5	9
L	2	5	0	4
C	6	9	4	7

Table A.1: Straight section stitch count, run of length 2

Length: 3	D	J	L	C
D	11	14	9	13
J	14	17	12	16
L	9	12	7	11
C	13	16	11	14

Table A.2: Straight section stitch count, run of length 3

Length: 4	D	J	L	C
D	18	21	16	20
J	21	24	19	23
L	16	19	14	18
C	20	23	18	21

Table A.3: Straight section stitch count, run of length 4

Length: 5	D	J	L	C
D	25	28	23	27
J	28	31	26	30
L	23	26	21	25
C	27	30	25	28

Table A.4: Straight section stitch count, run of length 5

Length: 6	D	J	L	C
D	32	35	30	34
J	35	38	33	37
L	30	33	28	32
C	34	37	32	35

Table A.5: Straight section stitch count, run of length 6

Length: 7	D	J	L	C
D	40	42	37	41
J	42	45	40	44
L	37	40	35	39
C	41	44	39	42

Table A.6: Straight section stitch count, run of length 7

Length: 8	D	J	L	C
D	47	50	45	48
J	50	52	47	51
L	45	47	42	46
C	48	51	46	50

Table A.7: Straight section stitch count, run of length 8

Length: 9	D	J	L	C
D	54	57	52	55
J	57	60	55	58
L	52	55	50	53
C	55	58	53	57

Table A.8: Straight section stitch count, run of length 9

Length: 10	D	J	L	C
D	61	64	59	62
J	64	67	62	65
L	59	62	57	60
C	62	65	60	64

Table A.9: Straight section stitch count, run of length 10

Length: 11	D	J	L	C
D	68	71	66	69
J	71	74	69	72
L	66	69	64	67
C	69	72	67	71

Table A.10: Straight section stitch count, run of length 11

Length: 12	D	J	L	C
D	75	78	73	76
J	78	81	76	79
L	73	76	71	74
C	76	79	74	78

Table A.11: Straight section stitch count, run of length 12

Length: 13	D	J	L	C
D	82	85	80	83
J	85	88	83	86
L	80	83	78	81
C	83	86	81	85

Table A.12: Straight section stitch count, run of length 13

List of Figures

List of Tables

Glossary

component A component is a section of a knot between from one crossing to the next. Standard components are: straight, quarter turn, point, and twist. 40

corner A run is bounded on both sides by a corner. There are four types of corners: D-type, C-type, L-type, and J-type. 44

C-type If the next dot after a run has a feature wall perpendicular to the original run, followed by a feature wall parallel to the original run, we call that corner C-type (Figure 3.12). 38

ch chain. 10

crossing A crossing is where a strand crosses itself or another strand. In a knot diagram, these occur at dots. A crossing may be over or under, referring to the position of the strand in question. Unless we're talking about which strand is on top, "crossing" is broadly interchangeable with "intersection", but we generally use "crossing" to emphasize the physical strands laid on top of each other, rather than simply their locations. 32, 34

D-type If the next dot after a run has no feature wall, we call that corner D-type (for "direct", Figure 3.9). 37

decrease Decreases (dec), unless otherwise noted, are worked as slst2tog. Usually we only work 2 stitches together at a time. However, if we need to decrease more (for example, if we have a segment with three stitches, and the next row calls for only one) we may have to work slst3tog, slst4tog, etc. See Figure 2.10. 10

feature wall A line drawn through a dot on the starting grid that redirects the strand, causing it to bend. 33

IC inner C-type corner (used in Python program). 52

ID inner D-type corner (used in Python program). 52

IJ inner J-type corner (used in Python program). 52

IL inner L-type corner (used in Python program). 52

increase Increases (inc), unless otherwise noted, are worked by putting multiple sc in the same stitch. Usually we only put 2 sc in one stitch. However, if we need to increase more (for example, if we have a segment with only one stitch, and the next row calls for 3 stitches) we may have to put more than one sc in one stitch. 10

intersection An intersection is where a strand crosses itself or another strand. In a knot diagram, these occur at dots. "Intersection" is broadly interchangeable with "crossing", but we generally think of an intersection as the *location* of a crossing. 6

IP inside point (used in Python program). 52

IQ inside quarter turn (used in Python program). 52

IT inside twist (used in Python program). 52

J-type If the next dot after a run has a feature wall perpendicular to the original run, and the dot after that has no feature wall, we call that corner J-type (Figure 3.12a). 38

L-type If the next dot after a run has a feature wall perpendicular to the original run, followed by another feature wall perpendicular to the original run, we call that corner L-type (Figure 3.12b). 38

OC outer C-type corner (used in Python program). 52

OD outer D-type corner (used in Python program). 52

OJ outer J-type corner (used in Python program). 52

OL outer L-type corner (used in Python program). 52

OQ outside quarter turn (used in Python program). 52

OT outside twist (used in Python program). 52

outline A row of slip stitches along each edge of a strand in a contrasting colour. 10

point A point component (Figure 3.6a) occurs when the section of the strand between two dots (intersections) encounters exactly two feature walls, perpendicular to one another. 40

quarter turn A quarter turn component (Figure 3.3b) occurs when the section of the strand between two dots (intersections) encounters exactly one feature wall. 40

run A straight portion of a strand that runs between parallel adjacent feature walls is a run. 36, 44

running stitch marker A running stitch marker is a length of scrap yarn in a contrasting colour to the main colour of the strand, worked back and forth through each row. It marks a line across the strand. Running stitch markers may be labelled or not. 6

S straight component (used in Python program). 52

sc single crochet. 10

segment A segment is the part of a strand between two running stitch markers. Sometimes one component is broken into several segments, because the component consists of different pieces worked independently of one another. 9

slst slip stitch. 10

slst2tog slip stitch two together. 10

straight A straight component occurs when the strand goes between two consecutive dots (intersections) without encountering any feature walls. 40

strand A physical knot would be made up of some linear solid, such as a rope; we call this a strand. Strand may refer to the line making up a knot diagram, or to the physical crocheted object. 6

subsequent row Every row after the foundation chain. 8, 9

twist A twist component (Figure 3.6b) occurs when the section of the strand between two dots (intersections) encounters exactly three feature walls, arranged in a U-shape. 40