

Algorithmic Pattern

Drawing Kolam Patterns in Stitches and Code

Anuradha Reddy Alex McLean

Then Try This

Published on: Mar 19, 2024

DOI: <https://doi.org/10.21428/108765d1.53f112a6>

License: [Creative Commons Attribution-ShareAlike 4.0 International License \(CC-BY-SA 4.0\)](#)

Kolam is a South Indian heritage folk art, a line drawing that creates winding, interwoven geometric patterns over a two-dimensional dot-grid structure. During a 10-day research residency (with [Anu Reddy](#) visiting [Alex McLean](#) in the [Then Try This](#) studio in Sheffield, UK), we collaborated on a series of kolam experiments building on the [Algorithmic Pattern](#) theme. We approached kolam from multiple perspectives: material, embodied, computational, structural, and rhythmic. In the following, we outline our insights from these experiments, hoping they inspire folks to explore heritage patterns in new ways. For us, an important aspect of traditions is how they are reinterpreted and remade as they are passed on.

We mainly work with kambi kolam, a type of kolam drawn in a single continuous line. For more background on this project, check out Reddy, Anuradha. 2023. “Kambi Kolam as Algorithmic Pattern.” In Algorithmic Pattern Salon. Then Try This. <https://doi.org/10.21428/108765d1.3bcd31e8>

We kickstarted the residency with a public-facing kambi kolam drawing workshop organised by [Pattern Club Sheffield](#). Anu gave the participants some background and context before inviting them to join her in creating kolams on paper.

If anyone wishes to draw kolams, here is an [instructional zine](#) Anu prepared for the workshop. It includes a link (via QR code) to a [p5.js sketch](#), a code that generates a grid template with a dots and lines structure that helps draw kolams.



Figure 1
Workshop participants at Pattern Club Sheffield



Figure 2
Workshop participants at Pattern Club Sheffield

Machine embroidery for handsewn kolams

Anu's main aim for the residency was to explore the possibility of hand-sewing kolam patterns supported by a Sashiko-inspired embroidery structure (Sashiko is traditional Japanese embroidery or stitching). By combining these two structures—looping kolam structures, which are normally drawn—with textile embroidery, we quickly encountered all the challenges of integrating two very different structures.

To meet these challenges we learned to think like our embroidery machine, using a thread to draw a single line around the fabric, with each stitch creating a structural bind point in our structure. We began by working directly in the free/open [Inkscape](#) software, specifying each stitch as drawn vectors and using the [Inkstitch](#) plugin to export the drawings into embroidery files (.pes) for sending to the embroidery machine (Brother NV2600). Inkscape includes useful facilities for duplicating drawn shapes following symmetrical patterns, but for full control of the arrangement and order of stitches, we quickly moved to automate the drawing using the free/open source [p5.js](#) creative coding software. There is a library available for creating embroidery files directly with the closely related Processing software called '[PEmbroider](#)', but we found it best to output our

stitch vectors using the SVG format, then use Inkscape/Inkstitch check and make final adjustments to the stitches, and export the embroidery file from there.

On the first attempt, we interwove the kolam thread in the inner stitches of the embroidered Sashiko ‘star’ by going under-over-under the machine stitches. However, as pictured, this did not hold the kolam curves well.

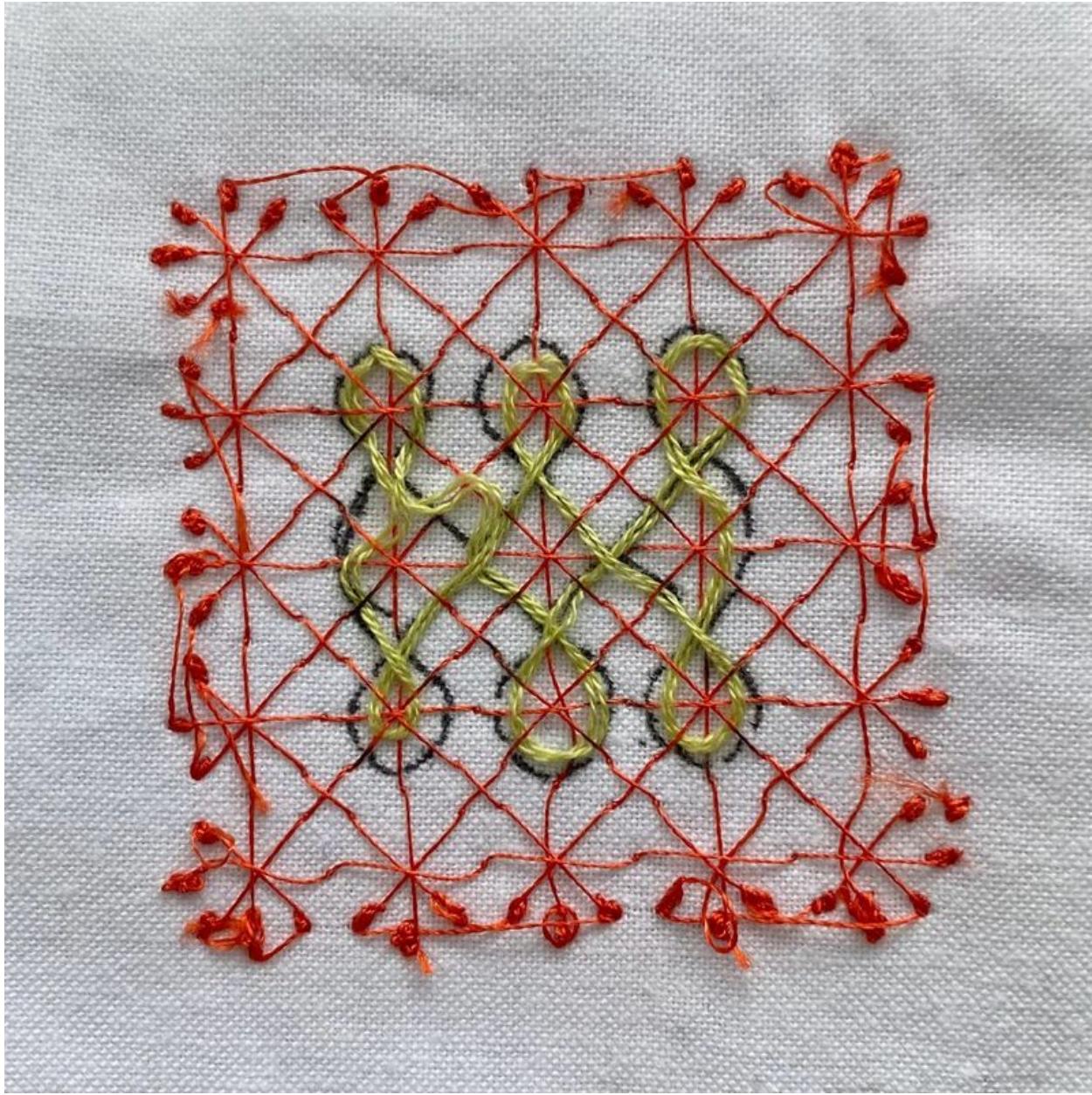


Figure 3

In our next attempt, we used the stitches outside the Sashiko star (crosses) to hold the curves, but the resulting kolam had sharp edges, resembling Lusona, a Kolam-like artform from Angola.

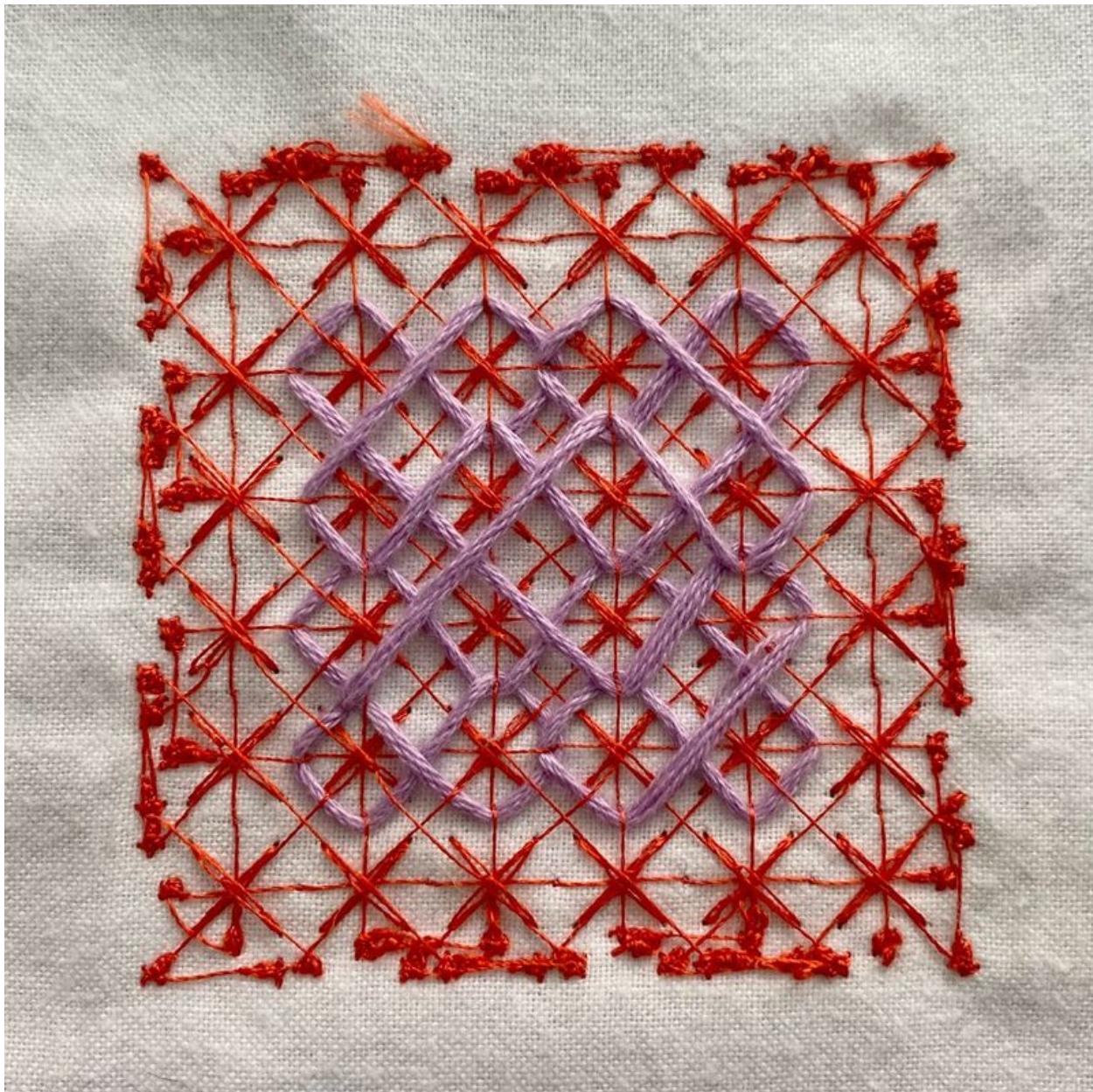


Figure 4

Anu then returned to a hand-sewn Sashiko stitch method as the base structure, which helped us understand which structural binding points the machine needed to reinforce to achieve a kolam.



Figure 5

At this stage, we also discovered that an $n \times n$ kolam is best achieved with a double-sized embroidered structure, i.e., $2n \times 2n$. We then adjusted the grid and sewing method to find a structure that worked well.

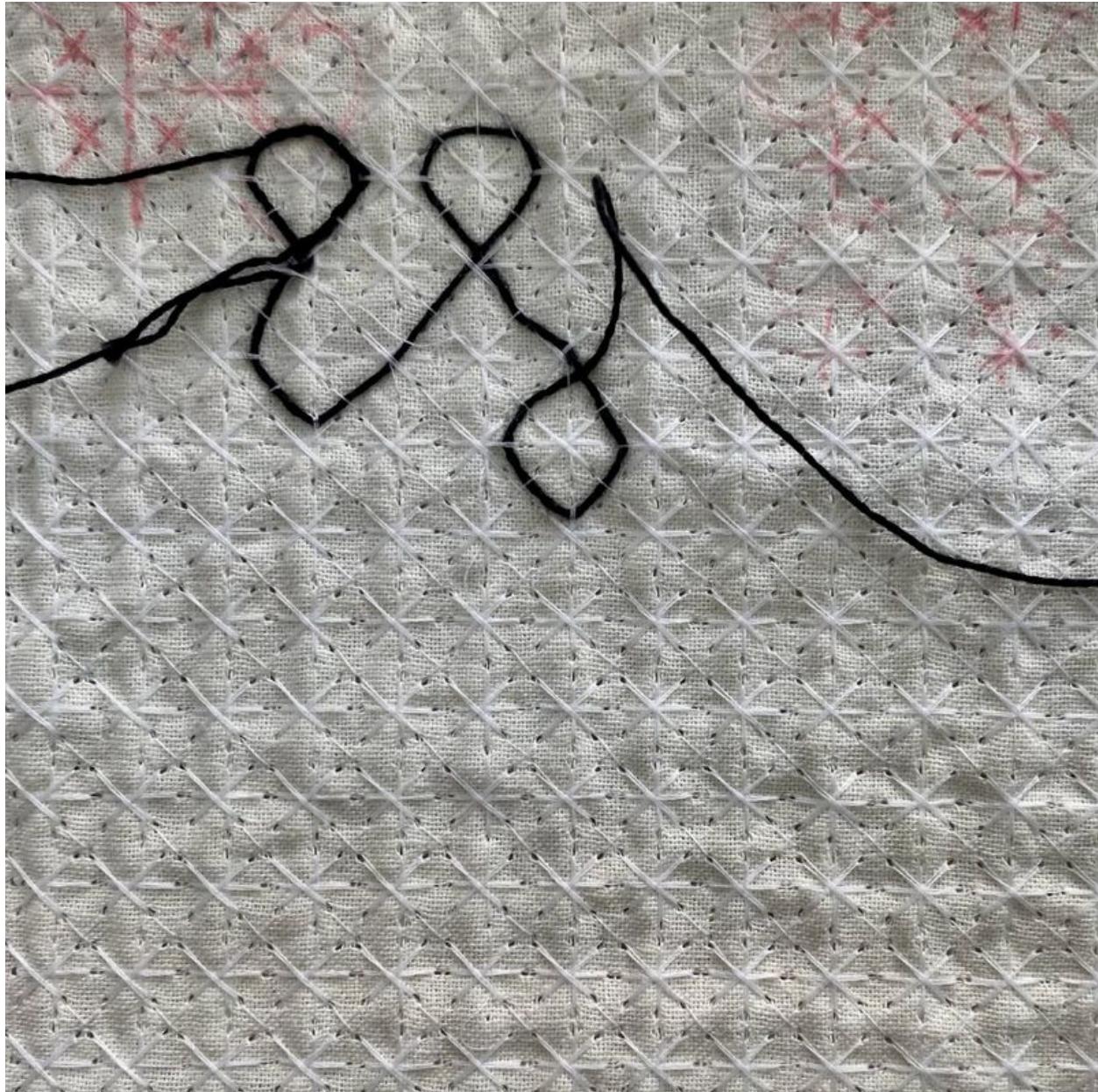


Figure 6

In the final iteration, we attempted to create a kolam as large as possible to fit the dimensions of the machine embroidery hoop and centre it on the next embroidery grid. For a 15x15 embroidered grid, we could thread a 7x7 Kolam ($n/2 \times n/2 - 1$), as shown below.

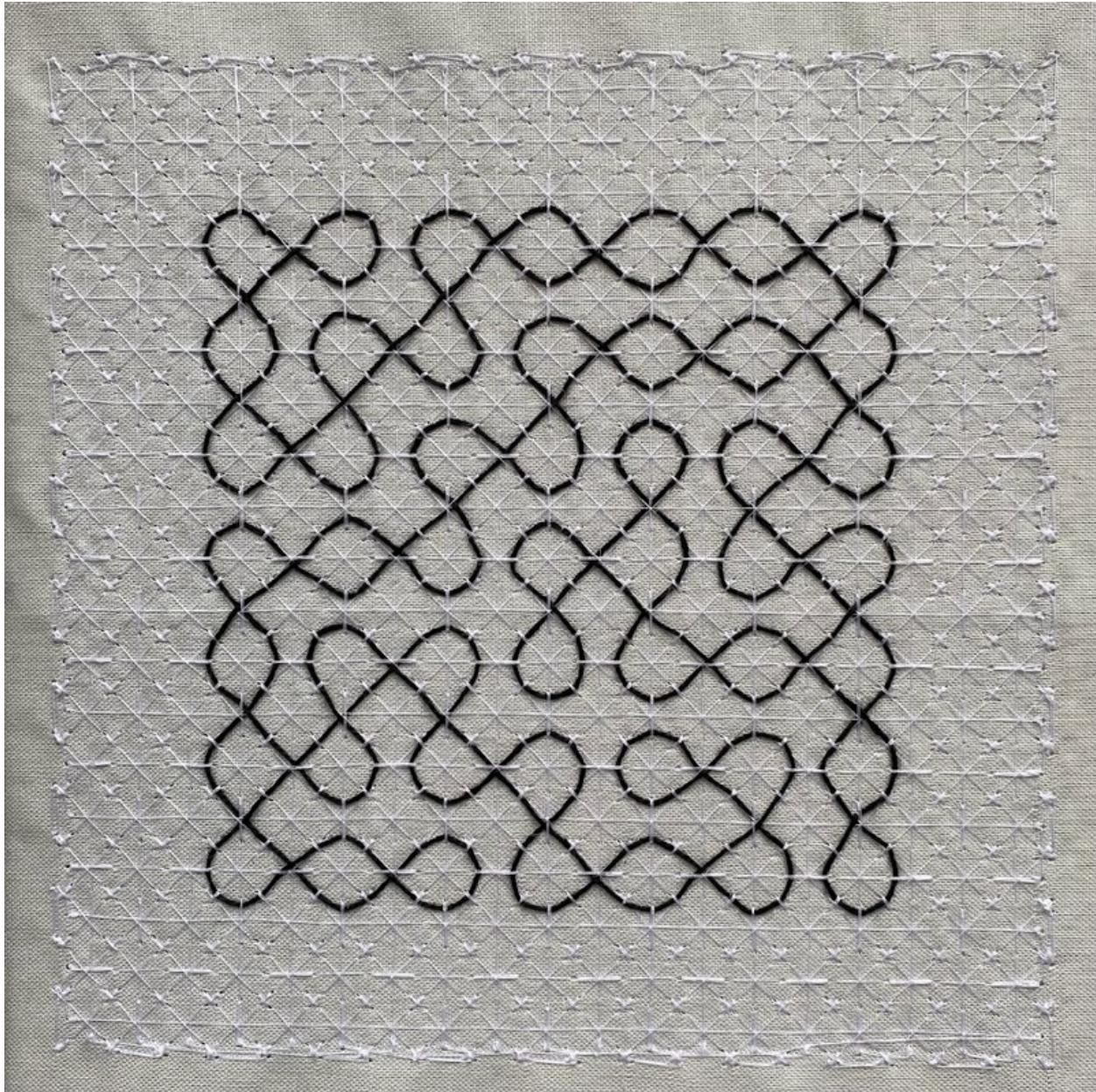


Figure 7

The final sketch is available here: [kolam support grid final](#). When you press the ‘play’ button to run the code, the stitch path will be shown on screen, but note that this does *not* show the full embroidery structure. The individual stitches aren’t visible, including where the lines are reinforced by going back on themselves to create three stitches in place (some of these stitches may be visible as slightly thicker lines). Furthermore although discrete diagonal, horizontal and vertical lines are shown, in practice our embroidery machine does not cut the threads at the ends of the lines, so they will be joined into a single line. We took care to draw the lines in the right order and direction to avoid long ‘floating’ stitches between them. The reason for separating

the lines is by default ink/stitch adds ‘lock stitches’ at the ends of lines; we found that it was important to reinforce stitches at the edges of the grid structure.

To save an SVG, change the indicated line 13 to set the saveSVG variable to true:

```
const saveSVG = true;
```

Then click the play button again, and a file grid.svg will be downloaded.

In inkscape with the inkstitch plugin installed, you will need to change some parameters, via the menu: Extensions > Ink/Stitch > Params. Change ‘Method’ to ‘Manual Stitch’, and click ‘Force lock stitches’ and click ‘Apply/Quit’.

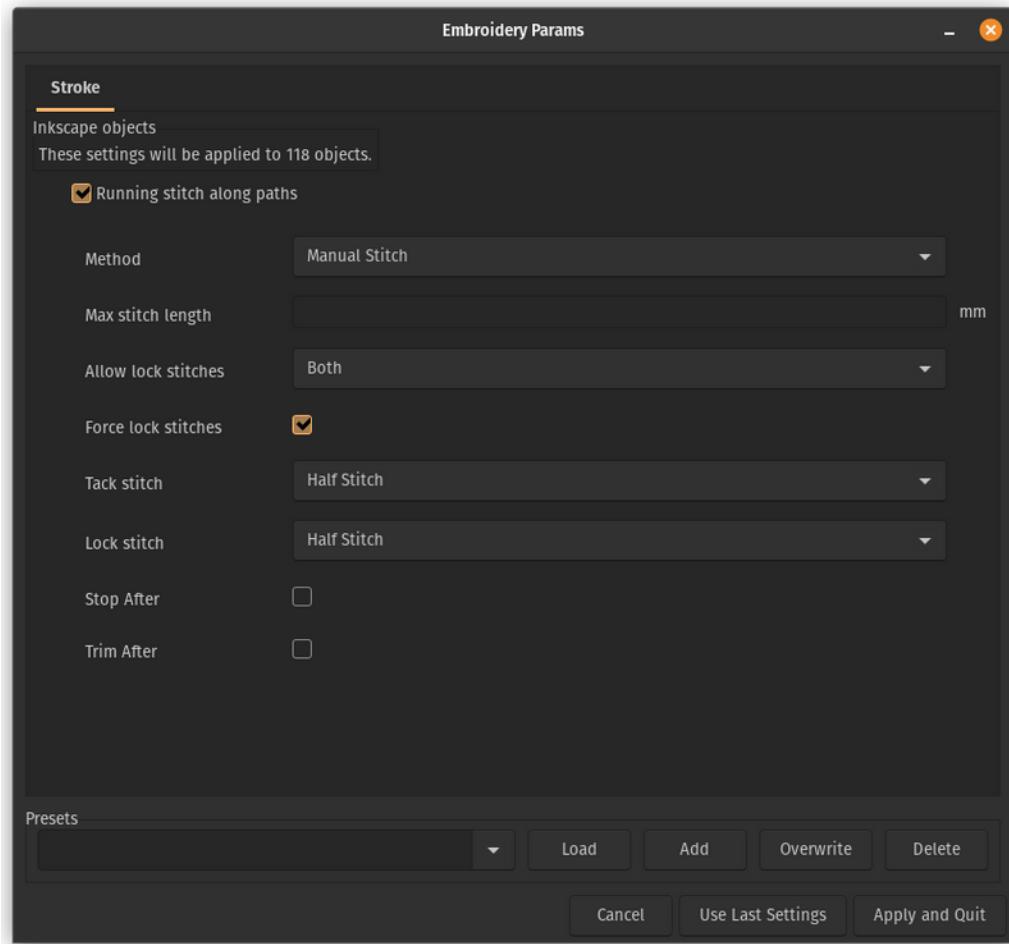
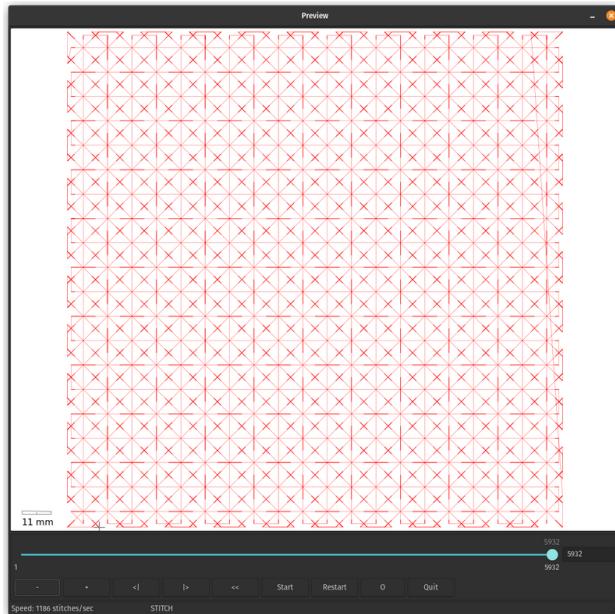


Figure 8

At this point you will also be able to explore the exact stitches that are made, via the preview window. Note that the design isn’t fully symmetrical - at this point we edited the stitches directly in inkscape, trimming off the edges of the design to create a fully symmetrical pattern.

**Figure 9**

Then the file can be saved as a .pes or other embroidery file format, for use on an embroidery machine. You can download the file we used here: [7x7-with-frame.pes](#).

Hand embroidery

There were clear trade-offs between machine and hand embroidery throughout this work. The embroidery machine could create large grids very quickly, but it can only perform the limited range of stitches possible, with its single needle that can only move up and down. For example, it would have been very useful to have been able to create a line of embroidered stitches that alternated between passing over and under previously embroidered threads. Unfortunately, embroidery machines can only stitch on top of what has gone before.



Figure 10
Toni Buckby (left) during a visit of Anu (right) and Alex to Toni's studio in Sheffield.

A highlight of the residency was visiting the studio of Sheffield-based textile artist and technologist [Toni Buckby](#), where we were exposed to the full range of possibilities of hand-stitching, as its own world of advanced craft technology. Toni's long practice and recent PhD research focus on blackwork embroidery, and she has deep knowledge of hand-sewing structures, including the possibilities of their integration with electronics (e-textiles). She introduced us to terminology and techniques around 'whipped stitches', which turns out to be what we are trying to achieve with hand-sewing kolam into machine-embroidered grid structures. Here the sewing structures and the structure of threads become interdependent, and Toni was able to advise on both the stitch paths and the use of single-strand pearl cotton to achieve the right result.

Automated (and therefore simplified) kolam drawing

Kolams are typically drawn with a set of hand moves or gestures. Alex wanted to create kolams in p5.js using Bezier curves and repeats for symmetry. He then automated kolam patterns by translating the hand gestures into simple computer commands. The first version of this script is embedded below.

Visit the web version of this article to view interactive content.

Script 1

First iteration of our Kolam drawing script, created in the p5.js environment.

The above script draws a kolam from the top left, using the following sequence of right (r), left (l) and straight (s) movements: "rssssrrsssllssllssllssrrssssrrrsllsrr" (the final r is omitted in the above drawing so that you can see where it begins and ends).

This made it possible to add another visual dimension to Kolam, in particular the Z dimension as a passage of time. This was reminiscent of Anu's earlier experimentation with creating physical kolams with wire.

Visit the web version of this article to view interactive content.

Script 2

Version of the kolam sketch where the time over which the kolam would be drawn is visualised using the third dimension.

The above shows a larger kolam, following the path "lllssssllssssrrssrsllsssslllsrrssssrrssrsllsrrss".

We found we could then create 3-dimensional kolam spirals, simply by repeating the instructions.

Visit the web version of this article to view interactive content.

Script 3

3D spiral Kolam drawing

Perspectives on turning

At this point, we re-evaluated how to describe a kolam path as a sequence. So far, we have been making and following sequences of left (l), right (r) and straight (s), which we can say is relative to the person (or other agent!) drawing it. This presents some conceptual difficulties though, as ‘left’ and ‘right’ only works if you are continually turning yourself and/or the paper as you draw. In practice, this makes it difficult to transcribe a kolam drawing as a sequence of left/right/straight movements without continually making mistakes.

One alternative we considered is to turn relative to the *previous* turn, an approach which Alex had previously taken [while drawing weaving structures](#). The aim here is to take the perspective of a thread itself, which doesn’t have a left or right-handedness (although it *does* have S/Z direction spin). In this scheme, “left left right right right” would become “turn out, turn in, turn out, turn in, turn in”, where ‘out’ means turn in the opposite direction to the last turn, and ‘in’ means in the same direction as the last turn. This felt a little bit easier as we didn’t have to continually turn a drawing while transcribing it; however, having to compare each turn with the previous one was still tricky, leading to some mistakes.

Using the turn-based path sequences to make kolams, we then realized that we did not have to specify the turn direction at all! This is because the turn direction can be inferred from the ‘straight’ movements; for each straight movement, the turn direction switches between left and right to turn around the correct grid point or dot according to the rules of kambi kolam. So, the only rules we need are ‘straight’ and ‘turn’, with the direction inferred from the context.

Strudel rhythms

We began exploring our kambi kolam visualisations within the [Strudel](#) environment, which is designed mainly for live coding algorithmic music patterns.

Press *play* below to run an example script. If the kolam isn’t visible, please [open the script in its own window](#), and press the play button there.

Visit the web version of this article to view interactive content.

Script 4

A Kolam drawn via the Strudel live coding environment. The kolam might not be visible in narrower browsers.

In the above, you can see that, as per the discussion in the previous section, the path is described using only straight (s) and turn (t) instructions. Each turn is sonified with a low ‘bass drum’ sound, and each straight with a ‘high hat’ sound. The kolam should be drawn smoothly, following a particular rhythm, where each sequence of consecutive turns or straights takes place within a single ‘beat’ (if the drawing or the sound is broken up or jerky, please try opening it in a different web browser). Adding this smooth yet rhythmic flow over time connects us more closely to the feeling of drawing the kolam by hand, opening up a great deal of ground for further experimentation.

Future work

We aim to continue exploring kolams via textile structures and strudel rhythms. Alex plans to add kolam sequences to a strudel-oriented Algorithmic Pattern tutorial series and to further explore machine embroidery. Anu wants to interweave a kolam over an e-textile conductive grid structure to make a capacitive sense touchboard. She also wants to try her hand at strudel to make her first live-coded tune.