Supplementary Material for
Algorithms to Construct Designs for Foundation Paper Piecing of Quilt Patchwork Layers

Anton Bakker[1] and Tom Verhoeff[2]

[1]Norfolk, VA, USA; `antonbakker.com`, `antonbakker30@gmail.com`
[2]Dept. of Math. & CS, Eindhoven University of Technology; `T.Verhoeff@tue.nl`

Overview

We provide

- more (computer generated) images of quilts with periodic tilings;
- some complete sewing plans for the blocks of such tilings;
- a description our version of the algorithm to determine whether a given mesh is FP pieceable, and
- a description of our optimized algorithm to make a mesh FP pieceable (if possible) using auxiliary cuts.

Patchwork designs

See Fig. 1 for more periodic tilings on quilts. Fig. 2 shows a (computer generated) quilt used as bed cover. Another design is shown in Fig. 3.

Figure 1: *More periodic tilings on quilts (computer generated images).*

Figure 2: *Bed cover with periodic tiling (computer generated image).*

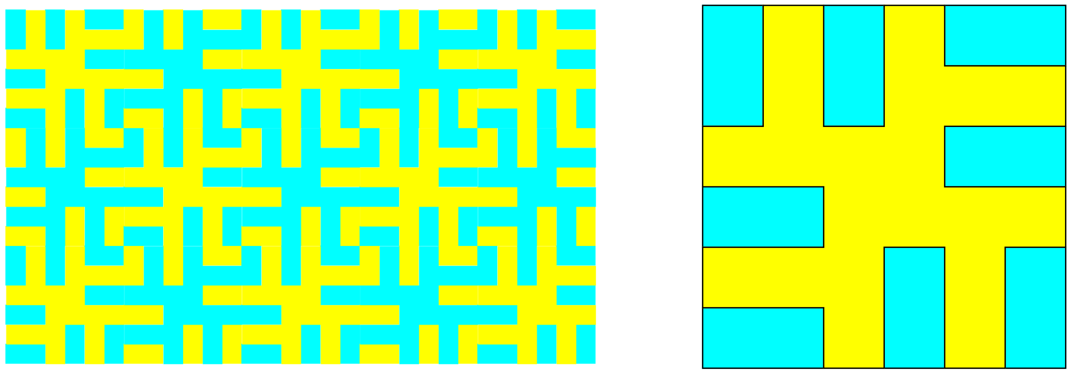

Figure 3: *Another periodic tiling (left) and a unit cell (right).*

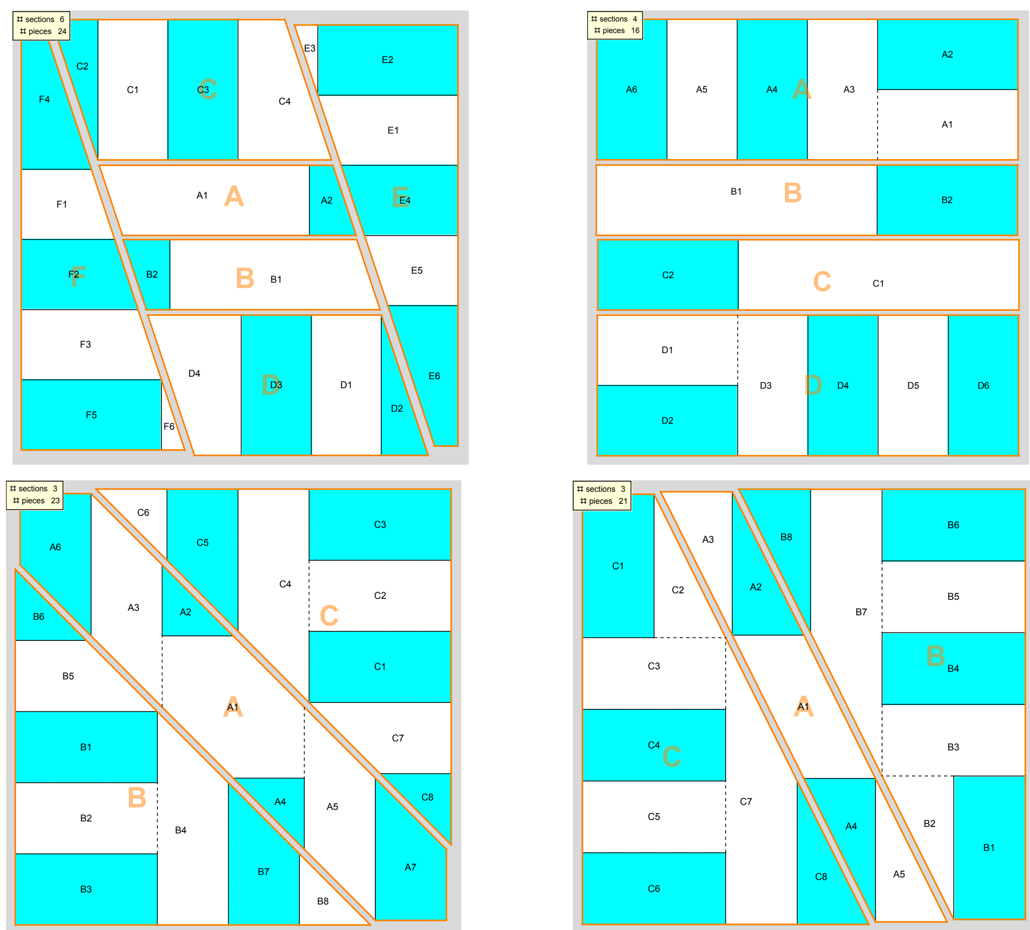

Figure 4: *Sewing plans: 30 pieces in 4 sections (top); 28 pieces in 5 sections (different unit cell*

Complete sewing plans

Fig. 4, top left, shows what our greedy algorithm (selecting two points on the convex hull) could give for the tile in Fig. 3, where none of the sections contain interior points of the mesh (12 interior points, hence, 6 sections with 24 pieces). Fig. 4, top right, shows a manually obtained sewing plan with 4 sections and only 16 pieces. The bottom row of Fig. 4 shows two sewing plans with only 3 sections where the section sewing plans were obtained by our optimizing algorithm. In the one on the right, the smallest piece is larger than on the left.

Fig. 5 shows a tile with a periodic tiling and a unit cell. Fig. 6 shows how two unit cells are combined into a (quilt) block, so that all blocks are the same (not only in terms of the sewing pattern, but also in terms of colors). Fig. 7 shows a complete sewing plan: how to assemble pieces into 6 sections (A through F), and how to assemble those 6 sections into a block. Alternative sewing plans are shown in Fig. 8

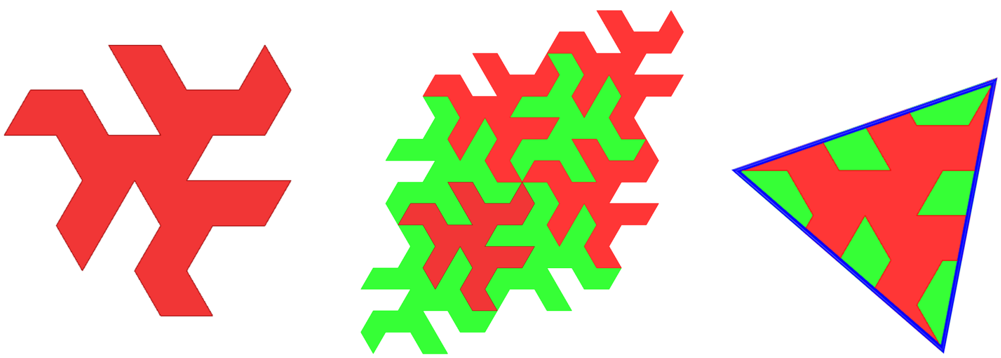

Figure 5: *Tile (left); periodic tiling (middle); a unit cell (right)*

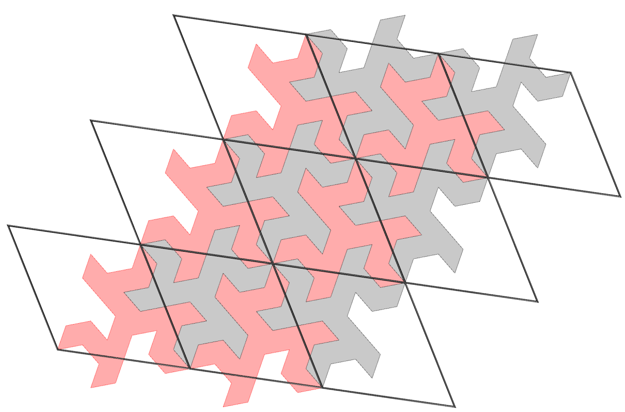

Figure 6: *Blocks of periodic tiling, consisting of two (other) unit cells*

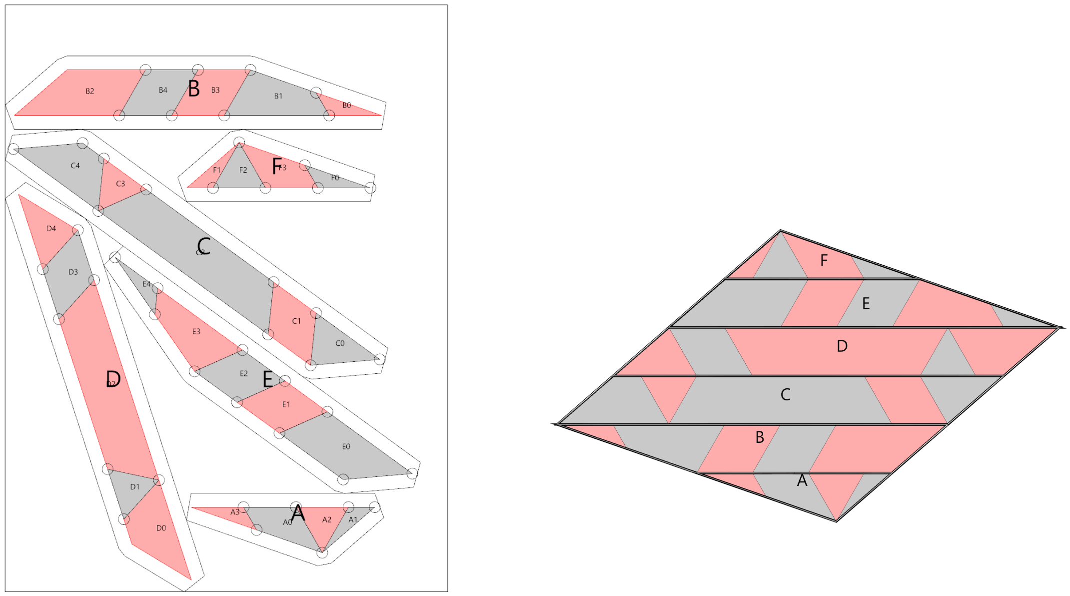

Figure 7: *Sewing plan: 28 pieces in 6 sections (left); overall (right)*

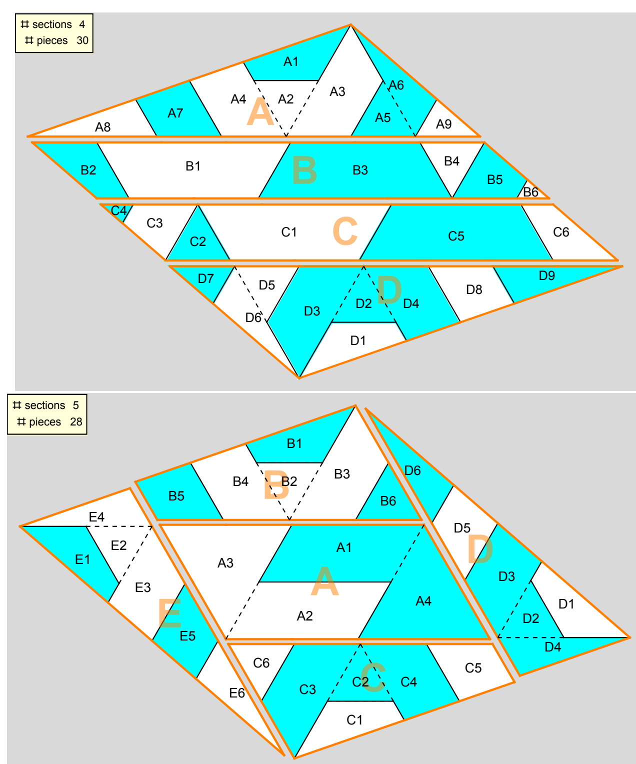

Figure 8: *Sewing plans: 30 pieces in 4 sections (top); 28 pieces in 5 sections (different unit cell)*

Algorithm for checking foundation paper pieceability of a mesh

Here we describe our algorithm to determine whether a mesh is FP pieceable 'as is'. The input of the algorithm is a mesh consisting of a collection of vertices and a collection of faces, where each face is defined by a list of its vertices in counterclockwise order. Some notes about meshes:

- For a face, the list of its vertices is interpreted cyclicly; rotating the list would not affect the mesh.
- For a face, all vertices incident on its edges appear in its list.
- In a mesh, all faces are attached along some of their edges into a single connected region. In (Leake et al., 2021), this region is called its realization.
- This region could have one or more holes (*maybe we should exclude this*), where fabric is missing, giving rise to multiple boundaries. Therefore a hole is not the same as a transparent piece of fabric, because there are no seams (stitches) on the boundaries.

From the input mesh we can determine the following entities:

- All *edges*, defined by cyclicly adjacent vertices of the faces.
- All *boundary edges*, i.e., edges to which a single face is attached.
- All *interior edges*, i.e., non-boundary edges.
- All *boundary vertices*, i.e, vertices on boundary edges.
- All *interior vertices*, i.e., vertices that are not boundary vertices.
- All *seams*, i.e., sequences of collinear interior edges.

The output of the algorithm is a boolean and a sewing plan (list of pieces in sewing order).

Algorithm steps:

1. If the mesh has only one face, it is FP pieceable: return True together with a singleton list of that piece (it will be the last piece assembled).
2. If the mesh has more than one face, then search for a (straight) seam across the entire mesh that separates it into two nonempty regions, at least one of which consists of a single face.
3. If no such seam exists, then exit with result False (the mesh is not FP pieceable).
4. If such a seam is found, then remove a single face region (there could be two), and prepend it to the list obtained by recursing at Step 1 after removing the piece from the mesh.

Algorithm for making a mesh foundation paper pieceable, if possible

Now we describe an algorithm that will make a mesh FP pieceable *by introducing auxiliary cuts*, if possible. The input of the algorithm is again a mesh. The algorithm returns a boolean (whether it was successful) and a list of pairs (sewing plan) consisting of

- a piece (to be assembled),
- an interior edge defining the cut (absent for the last piece in the list).

Algorithm steps:

1. If the mesh has only one face, it is FP pieceable: return True together with a singleton list of that piece (it will be the last piece assembled) and no interior edge.
2. If the mesh has more than one face, consider all *interior edges with two boundary points* (this is then a seam and avoids introducting unnecessary auxiliary cuts). If a boundary-boundary edge exists whose extension cuts off single piece, then proceed at Step 5.
3. If no boundary-boundary edge works, then sort all the *interior edges with one boundary point* by the size of the piece cut off by the edge's extension (using infinity if more than one piece is cut off). If the first edge indeed cuts of a single piece (in which case that is the *smallest* single piece that can be cut off) then proceed at Step 5.
4. If no edge is found, then it is impossible to make the mesh FP pieceable by introducing auxiliary cuts, and return False.
5. Prepend the single piece found with the cutting edge to the list obtained by recursing at Step 1 with the mesh from which that piece is cut off.

Notes:

- Edges are directed and are considered in both directions, to consider the two sides that can be cut off as a single piece.
- Interior edges without a boundary point will never be used. Even if they would cut off a single piece, they are never needed, because if you can find a sewing plan using them, then there also exists a sewing plan without them.
- Cutting off the smallest piece paradoxically helps avoid small pieces. Why? Because if you choose an alternative cut that cuts off a larger piece, then that smallest piece can only become smaller later on. See Fig. 9.

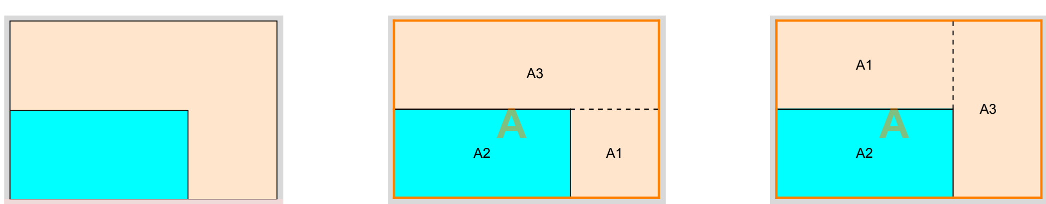

Figure 9: *Mesh (left); sewing plans with larger piece A3 cut off (middle) and smaller piece A3 (right)*

Cutting a mesh into sections that are FP pieceable

The last algorithm we describe is that for cutting a mesh into sections that are each FP pieceable. The algorithm takes as input a mesh and produces as output a FP sewing plan *for sections* that are each FP pieceable. That is, the completed sections can be stitched together following the rules of FP piecing. So, this gives rise to a two-level sewing plan. The algorithm's overall structure is the same as that for attempting to make a mesh FP pieceable. The main difference is that this algorithm always succeeds.

Algorithm steps:

1. Consider all cuts through pairs of mesh vertices passing through the interior, i.e., that cut the mesh into two nonempty regions.

2. Take one that chops off the "best" FP-pieceable section. Best could for example be: with largest area, or with least added auxiliary cuts, or whose smallest angle is largest.

   Such a cut can always be found, viz. by cutting along the convex hull of interior vertices.

3. Recursively proceed with Step 1 after removing the selected section from the mesh.

Note that the collection of candidate cuts can be a large (quadratic in the number of vertices).

To illustrate this algorithm and provide insight in how to optimize it, we use the mesh shown in Fig. 10. It has 60 vertices, of which 42 are interior vertices. The algorithm described in the paper would result in at most $\lceil 42/2 \rceil + 1 = 22$ sections. This improved algorithm considers $\binom{60}{2} - \binom{60-42}{2} = 1617$ vertex pairs. Each of these gives rise to two parts, for which we need to check whether they can be made FP pieceable. Altogether, the make-FP-pieceable algorithm is executed 3234 times.

Fig. 11 (left) shows all 102 cuts that chop off an FP-pieceable section. We see that they satisfy a monotonicity property. This makes it possible to rotate a sweep line around the mesh to traverse all *largest* FP-pieceable sections more efficiently. Fig. 11 (right) shows a sewing plan obtained by each time chopping off a section with the largest area. It has 59 pieces in 6 sections.

Fig. **??** shows four, more symmetric, plans with 6, 8, and 10 sections. But to obtain these, one cannot use the greedy approach based on largest area. Also pay attention to the number of pieces and the size of the smallest piece in each plan.

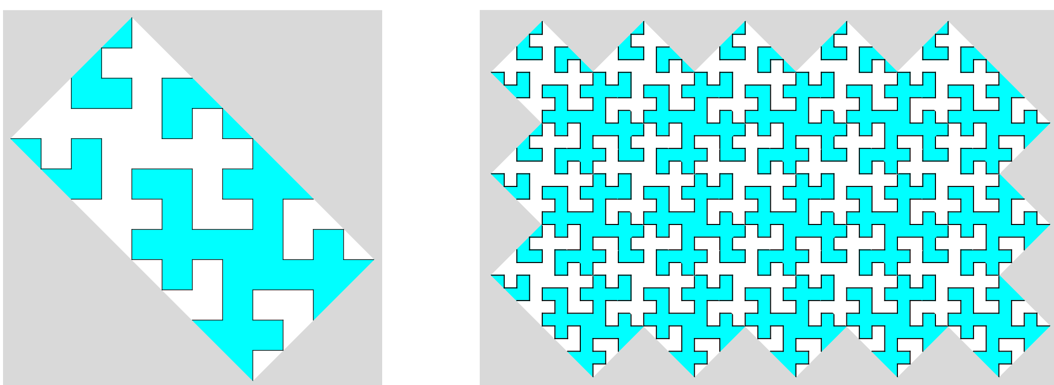

Figure 10: *Mesh of unit cell (left) and part of a tiling (right)*

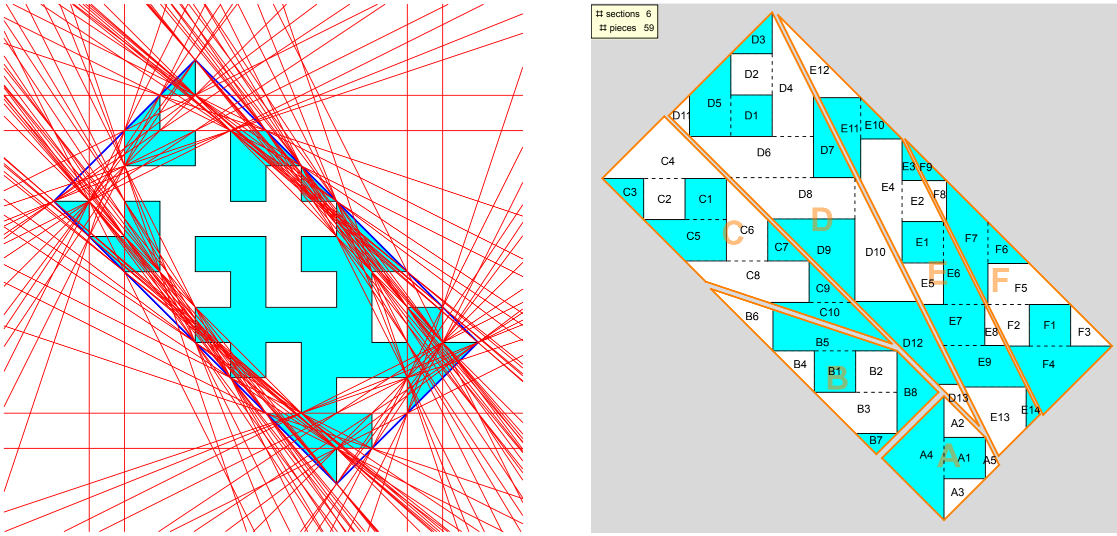

Figure 11: *All 102 cuts for mesh in Fig. 10 that chop off an FP-pieceable section (left); sewing plan (right)*

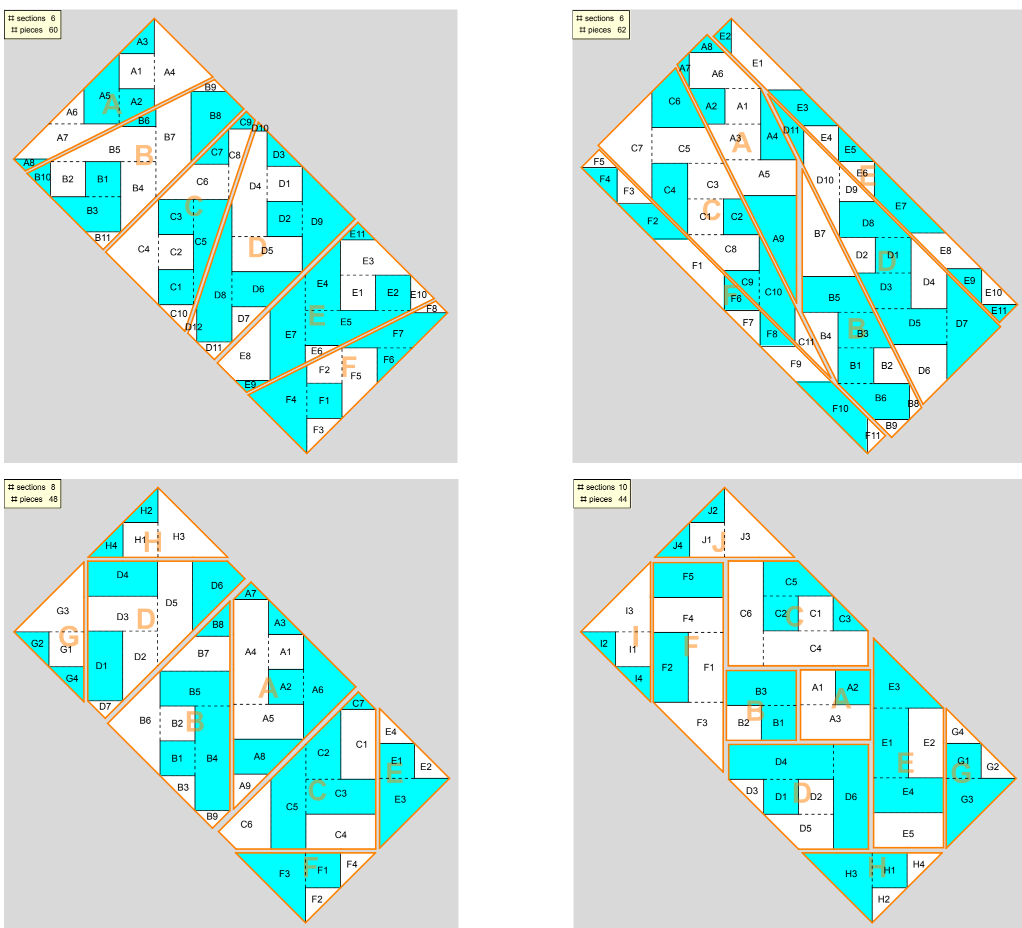

Figure 12: *For more sewing plans*