

# Algorithmic Quilting Pattern Generation

**Yifei Li**

Carnegie Mellon University

**David E. Breen**

Drexel University

**James McCann**

Carnegie Mellon University

**Jessica Hodgins**

Carnegie Mellon University

Graphics Interface 2019



# Background

Motivation

Related Work

Overview

Method

Fabricated Results

Discussions

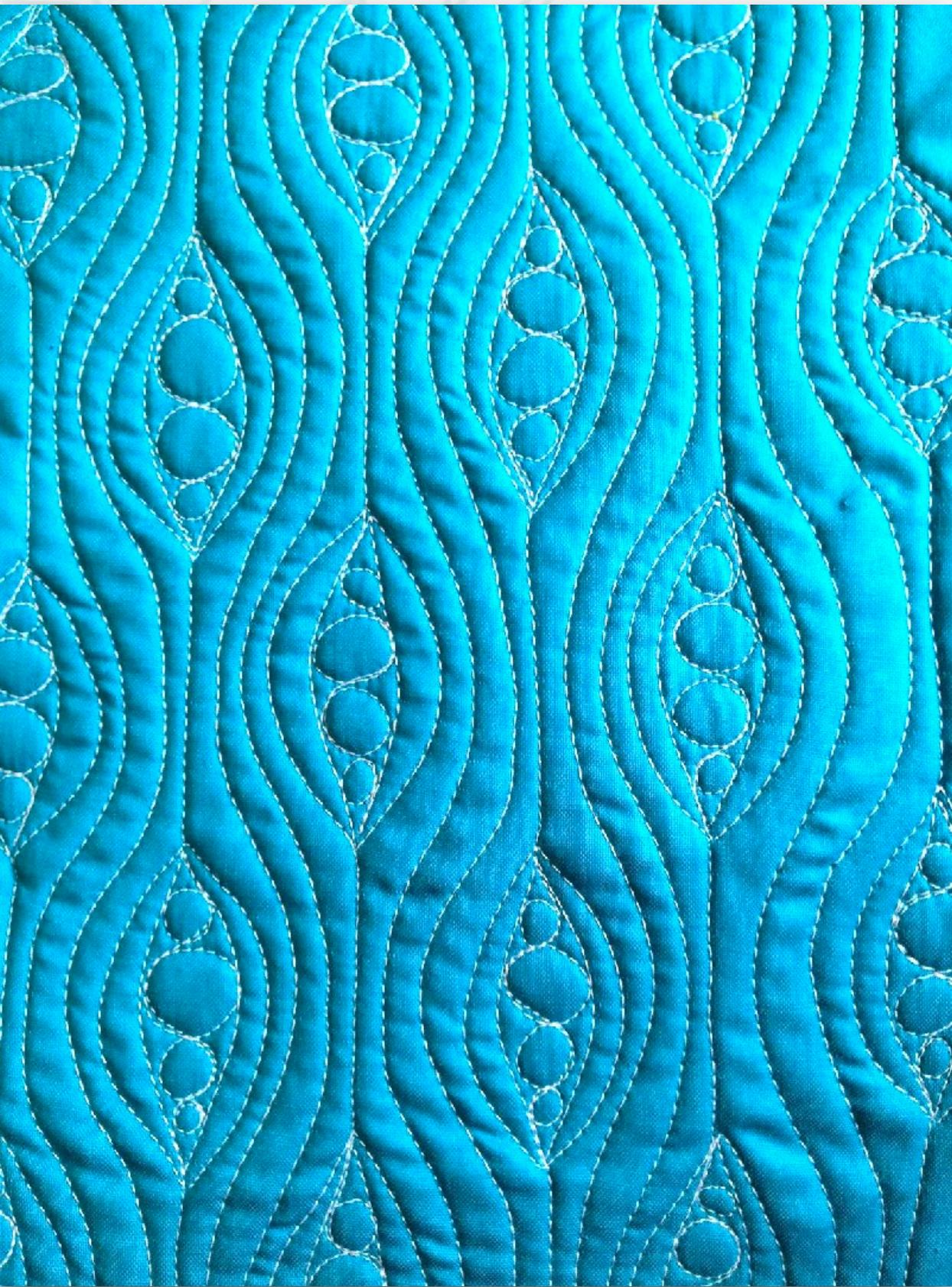
# Free-motion Quilting Patterns



**Bear Claws**

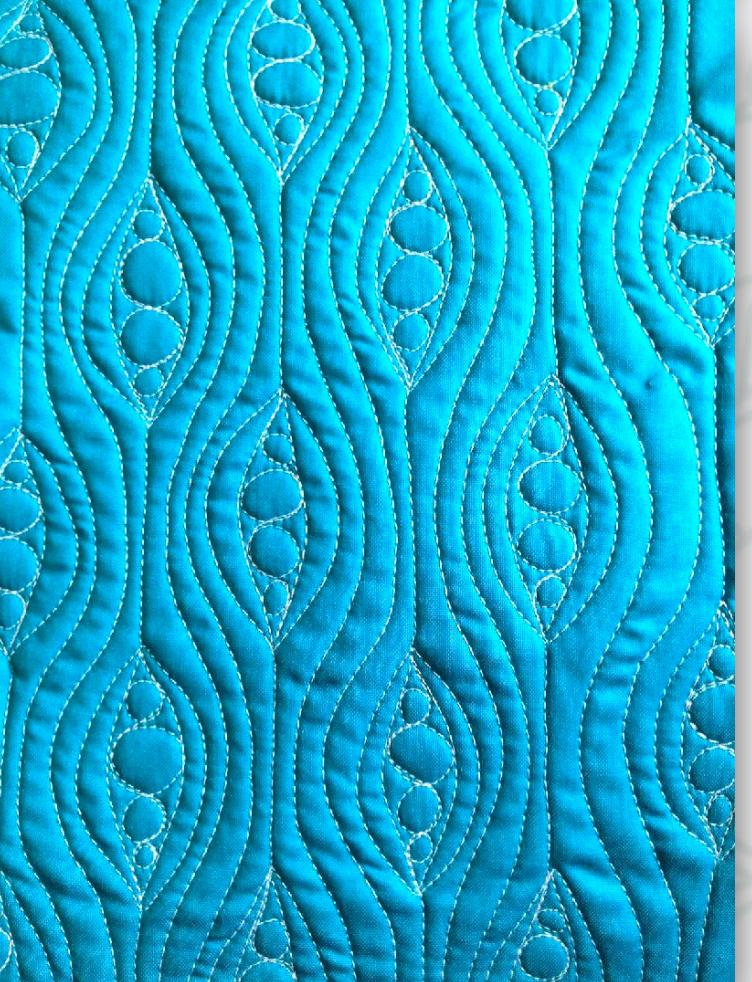


**Wibbly Wobbly**

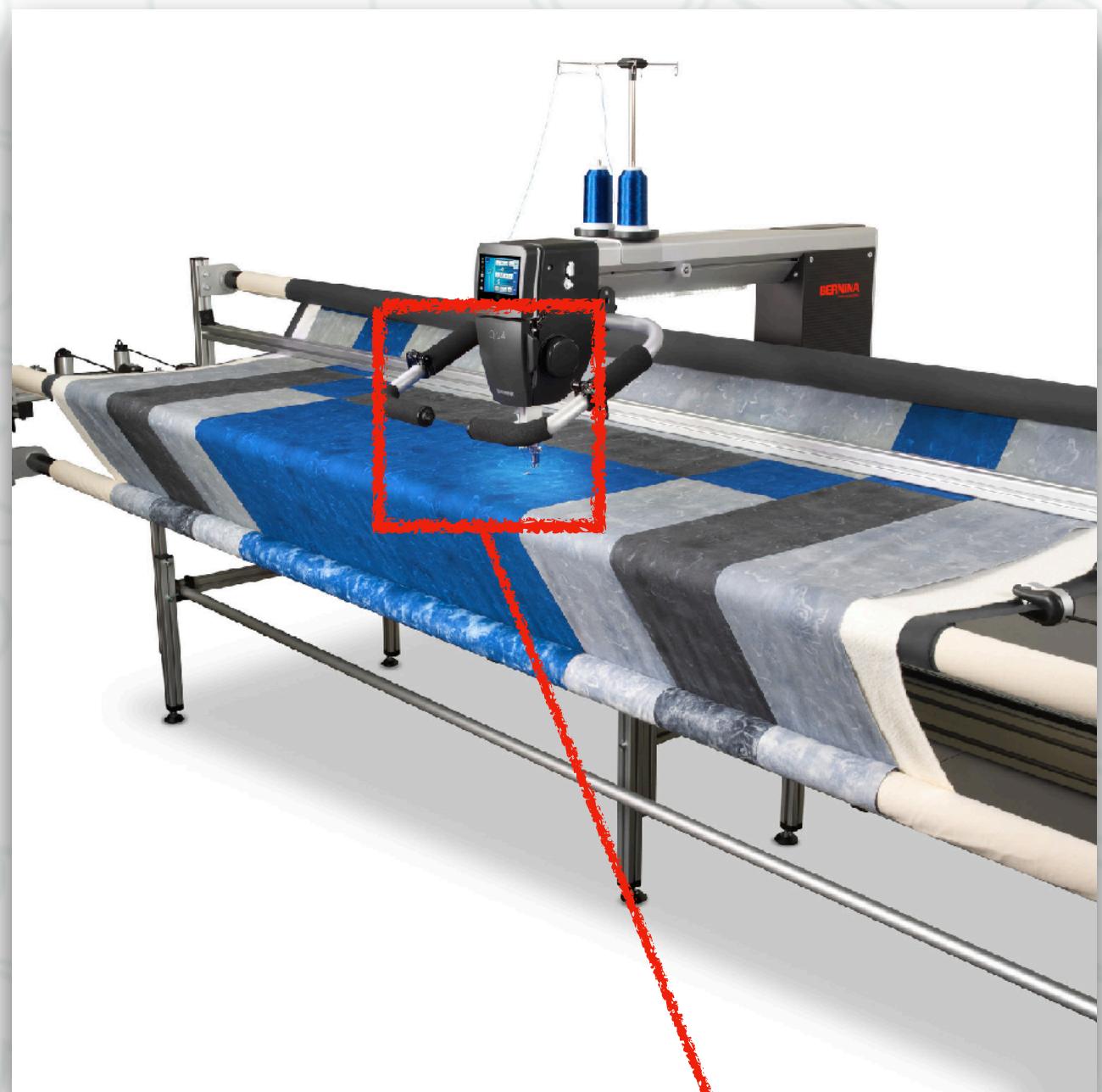


**Peapods**

# Free-motion Quilting (FMQ)

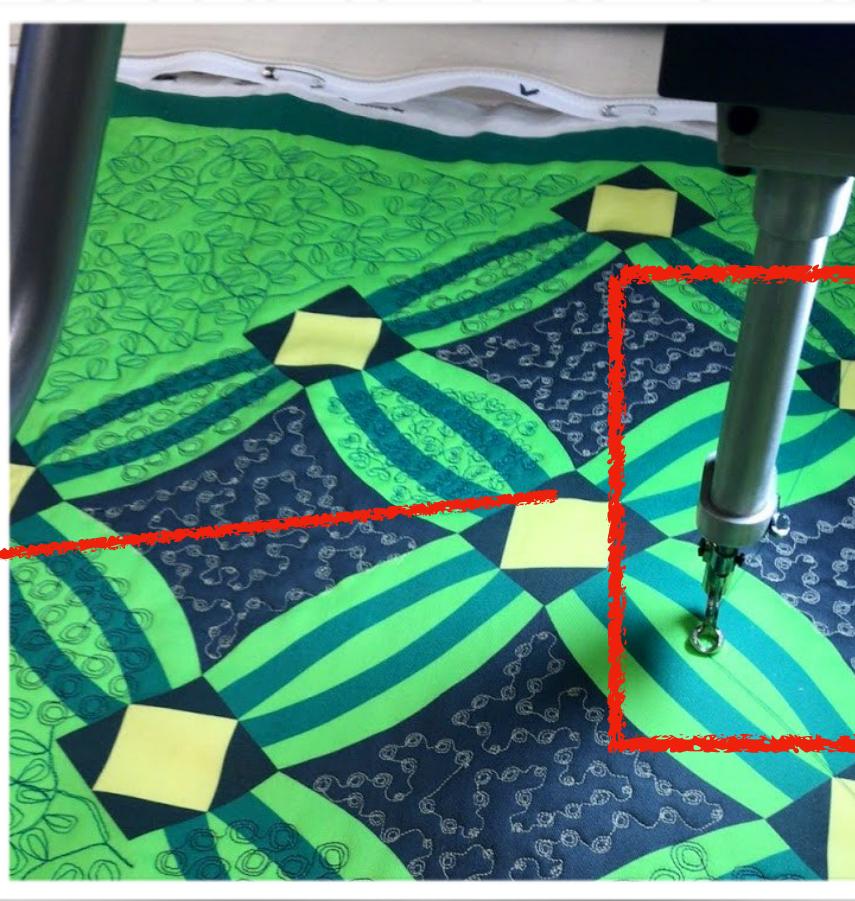


# Long-Arm Machine Quilting



Capable of quilting over an entire quilt top using computer control

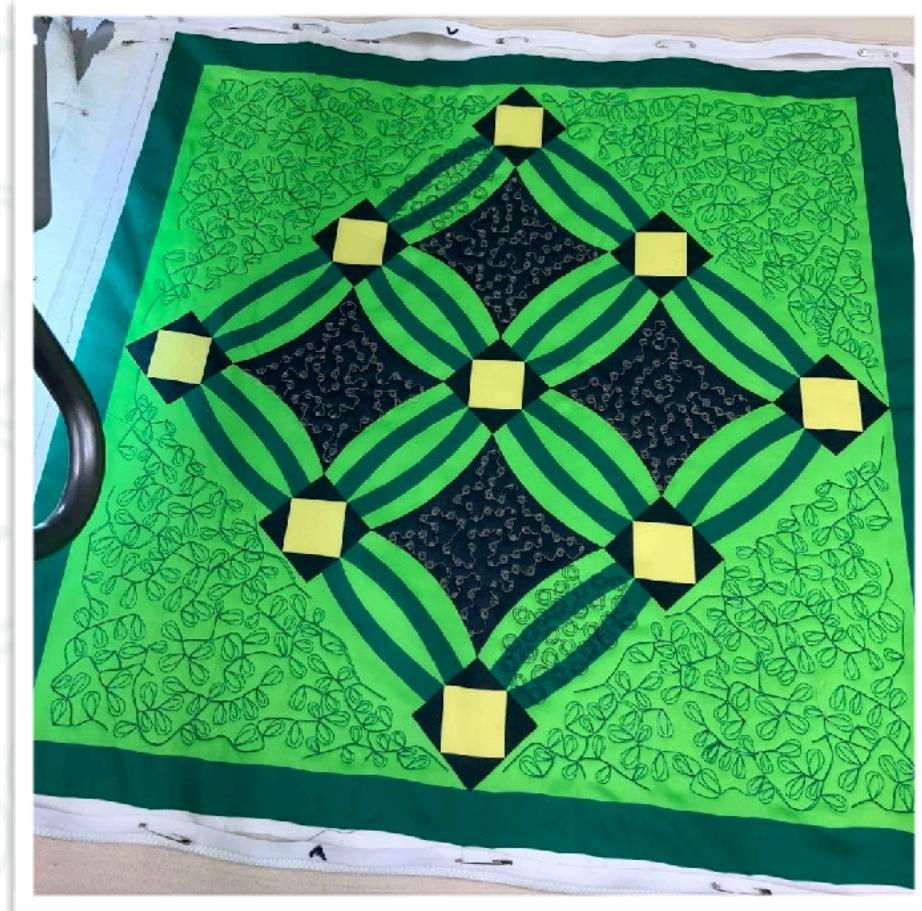
quilting head



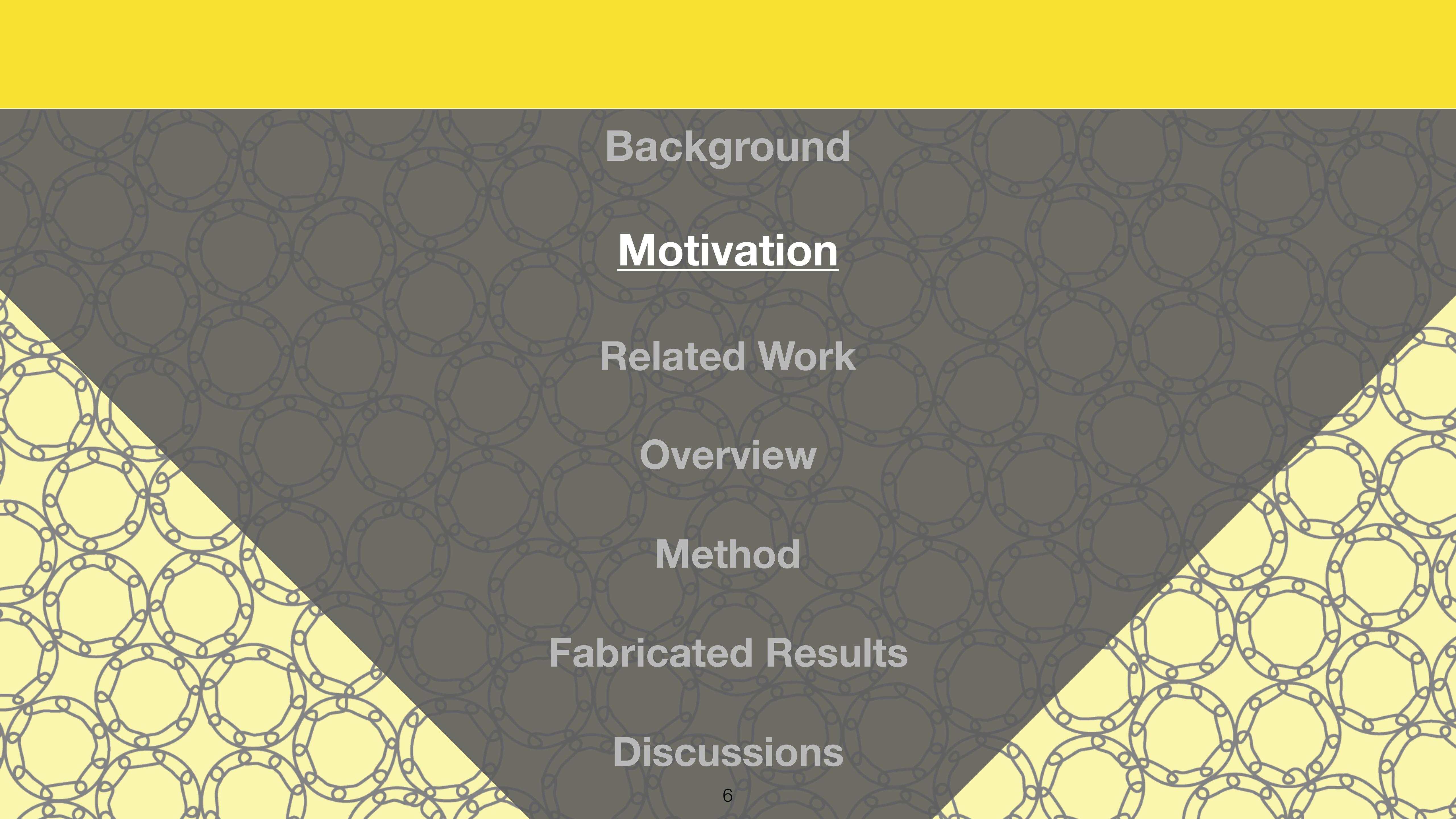
User manually aligns the quilting head to the starting point on the quilt



User inputs a digital pattern in the form of a 2D path to the computer control



Machine quilts the input digital pattern. User manually finishes quilting



Background

Motivation

Related Work

Overview

Method

Fabricated Results

Discussions

# Motivation

## Goal

Generate common free-motion quilting patterns that can be input to long-arm quilting machines

## Impact

Simplifies the process of pattern design and fabrication for experienced quilters

Naive quilters can make beautiful quilts using a long-arm quilting machine using our system

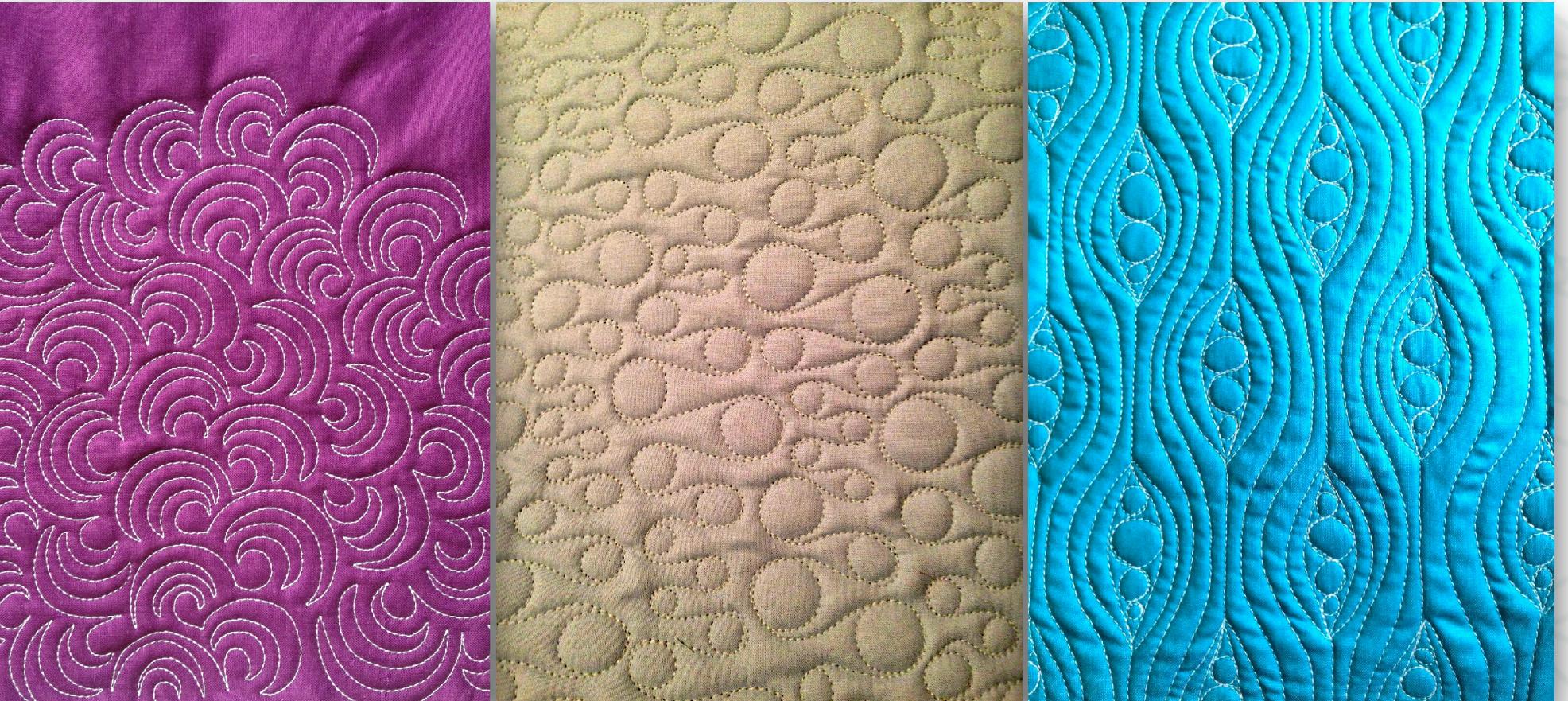
## Constraints

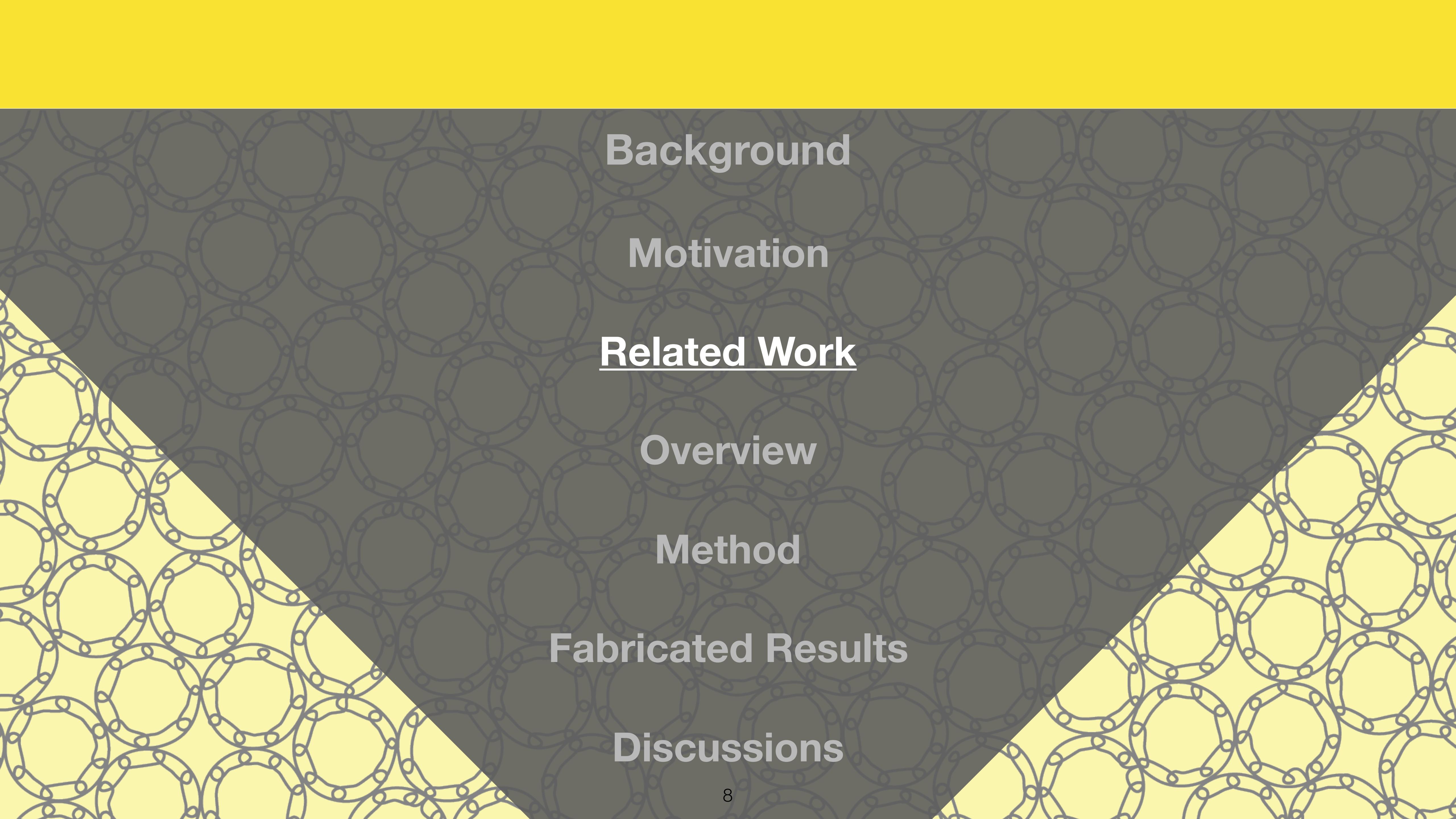
Pattern should consist of a single line path for large regions

- Reduce manual effort for starting and ending quilting

Stitches must be dense enough to hold the three layers of the quilt together

- Stitch path should be distributed roughly uniformly over the quilt with a minimum density





Background

Motivation

Related Work

Overview

Method

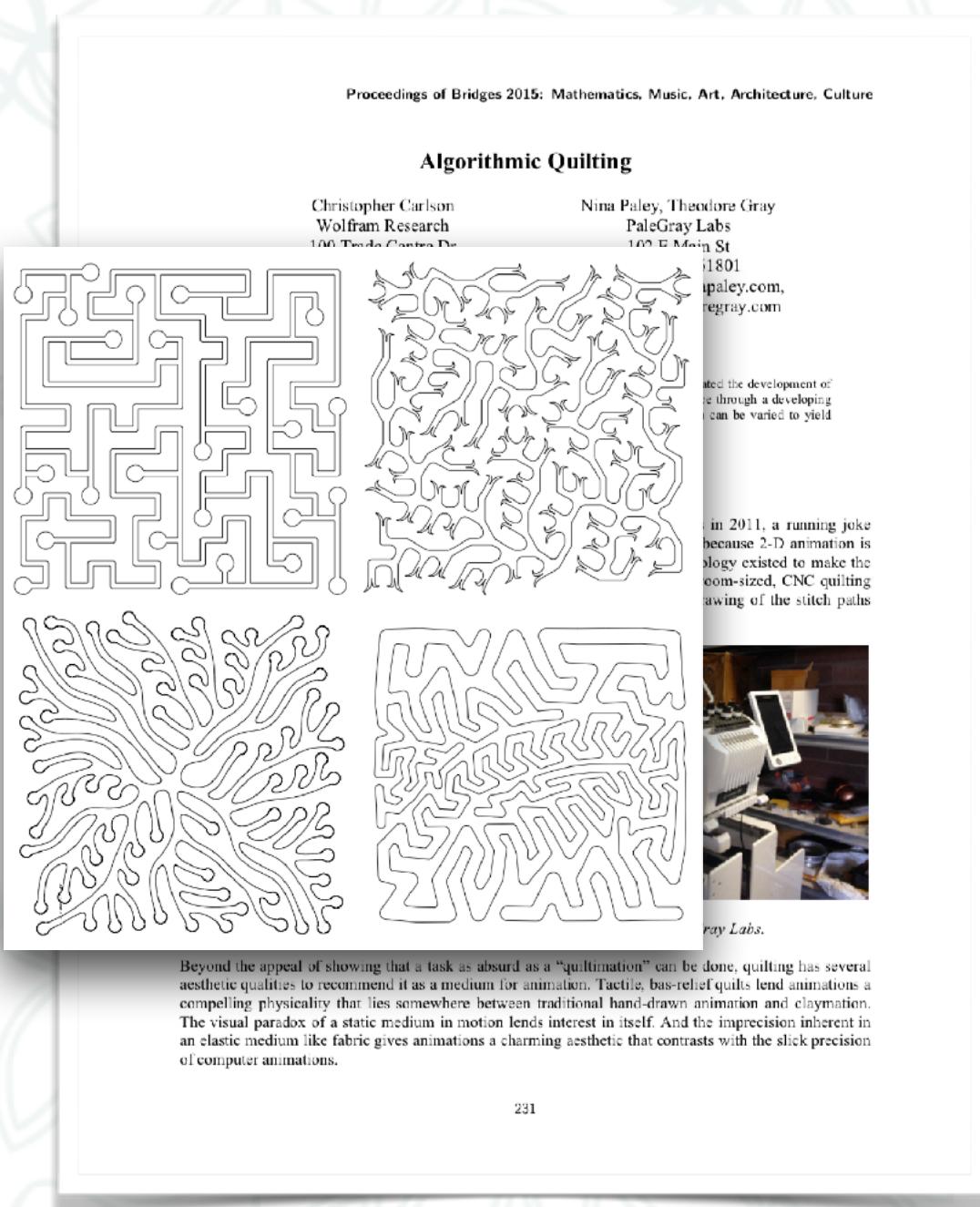
Fabricated Results

Discussions

# Related Work

# Algorithmic Quilting [Bridges 2015]

# Carlson and Gray



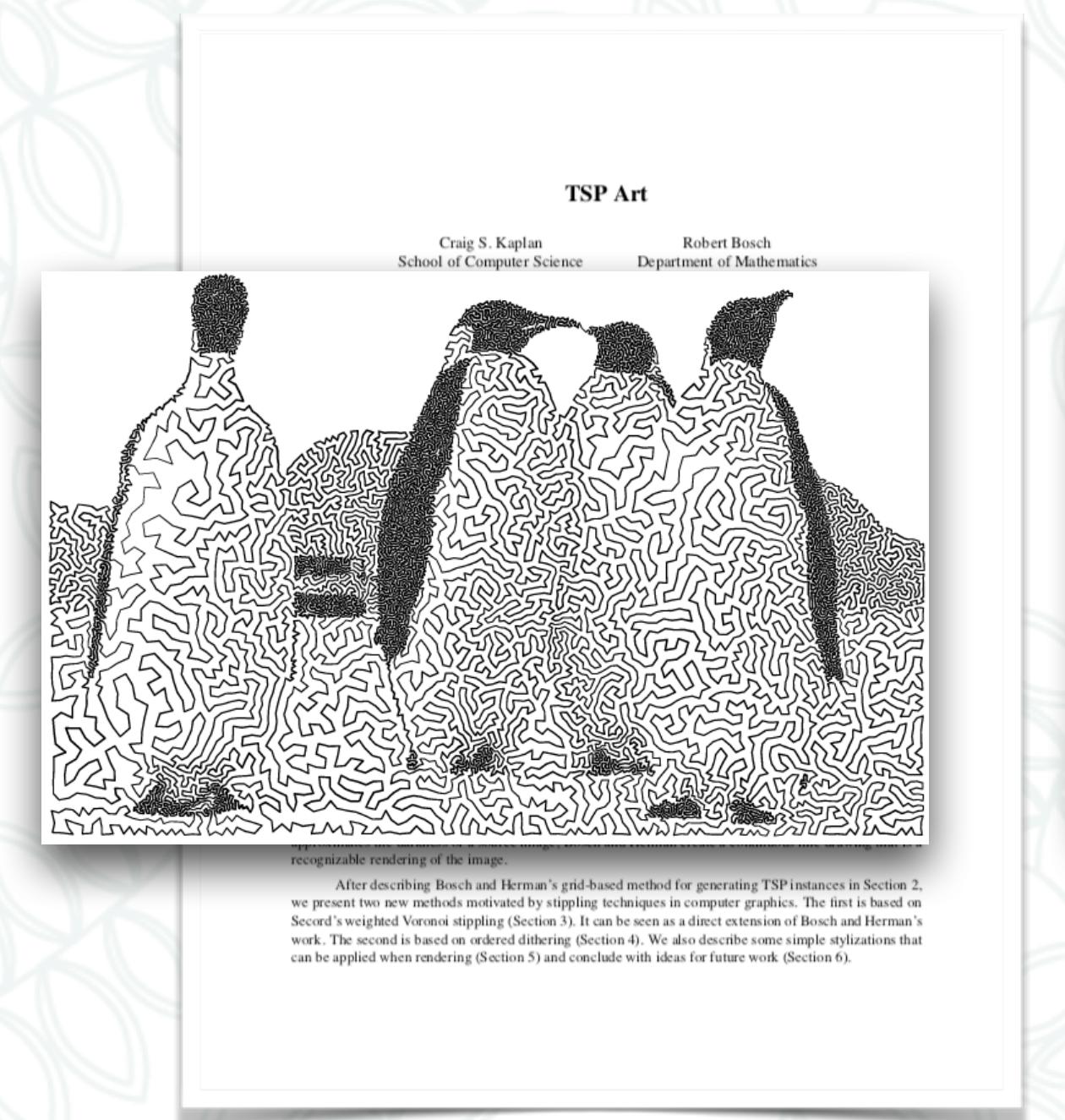
# Whole-Cloth Quilting Patterns from Photographs [NPAR' 2017]

Liu et al.



# TSP Art Bridges 2005]

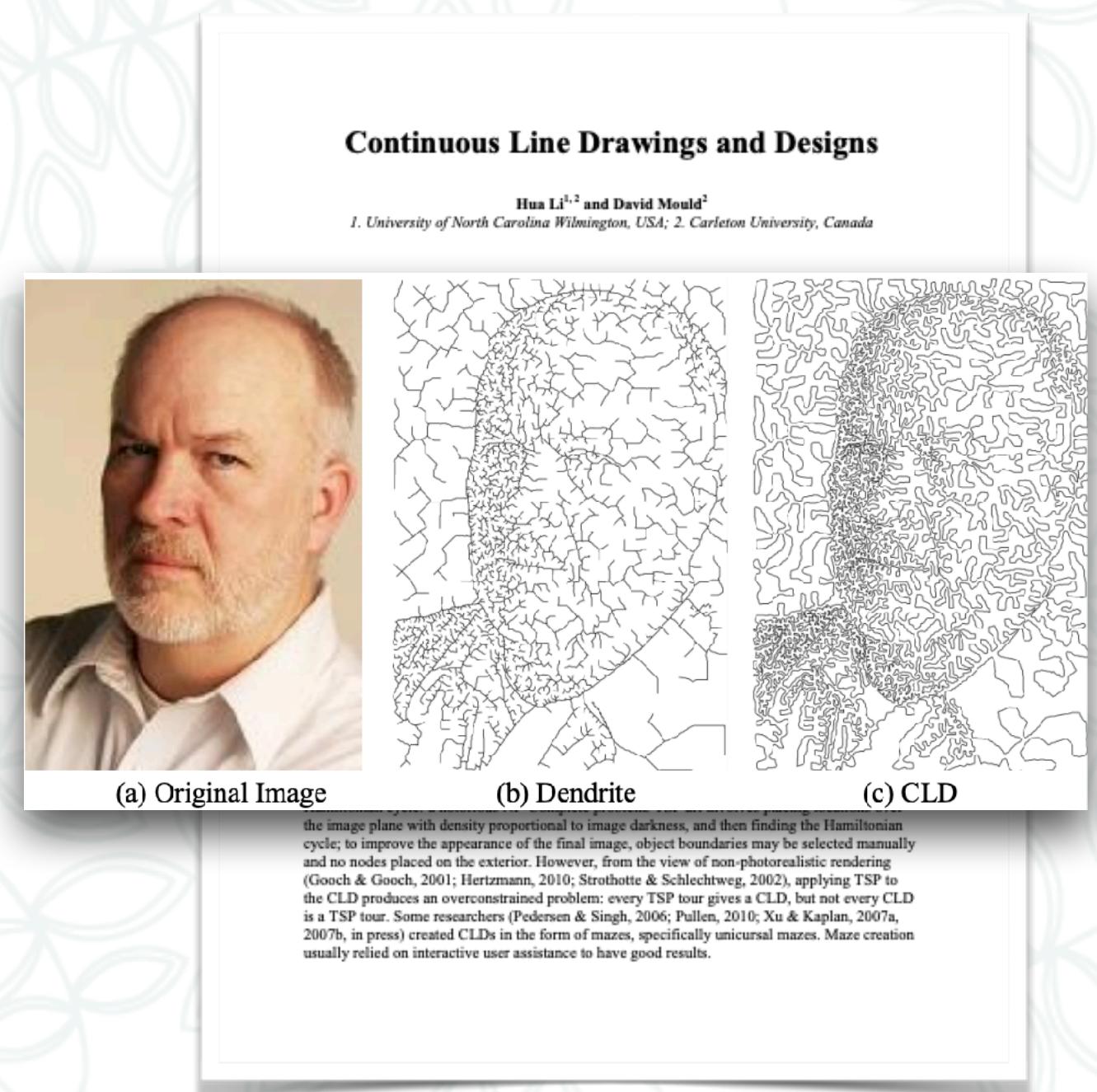
# Kaplan and Bosch

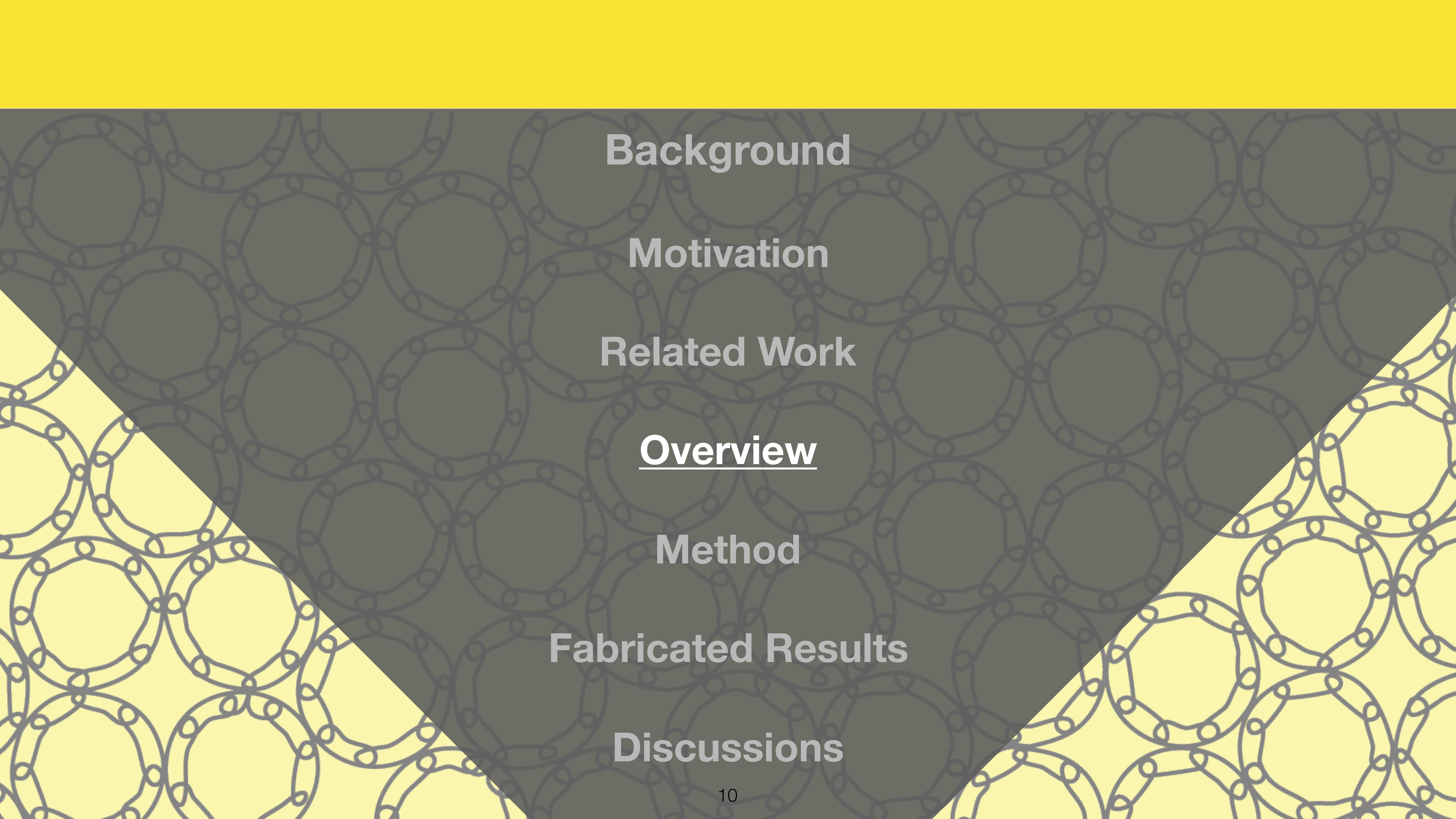


# **Continuous Line Drawings and Designs**

## **[International Journal of Creative Interfaces and Computer Graphics 2014]**

# Li and Mould





Background

Motivation

Related Work

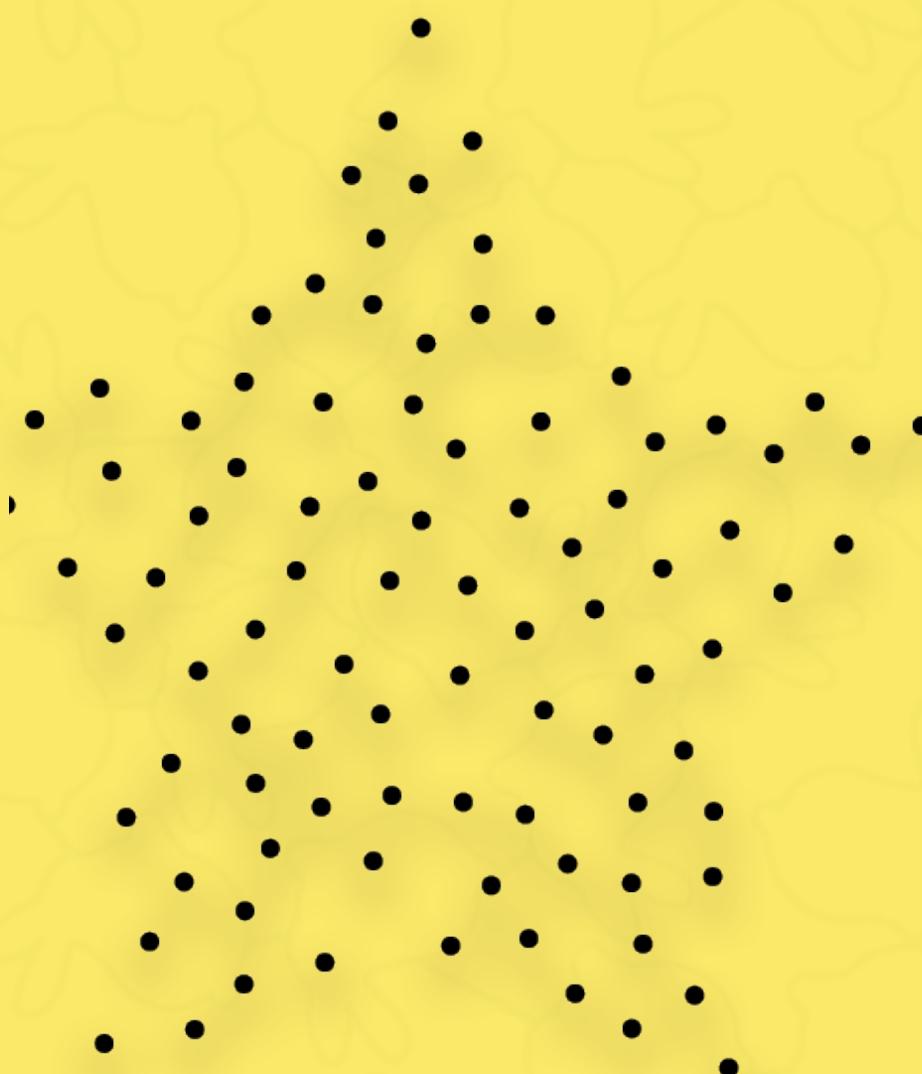
Overview

Method

Fabricated Results

Discussions

# General Overview



**Sample points**



**Stitching order via  
spanning trees/TSP tours**

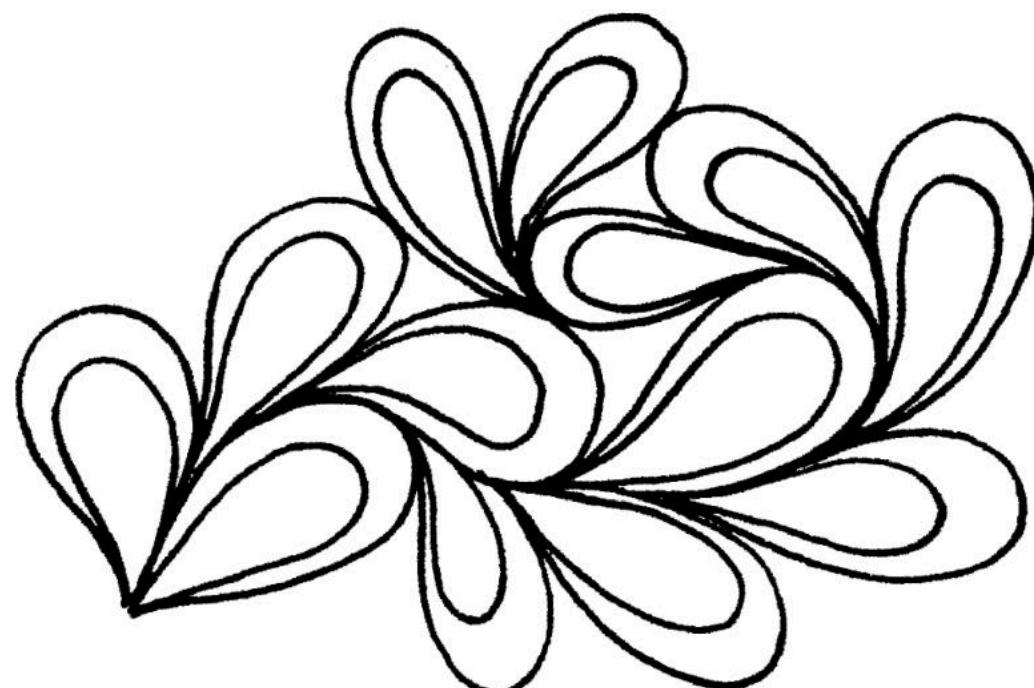


**Combine with decorative  
primitive to produce different  
quilting patterns**



**Quilt on a long-arm quilting  
machine**

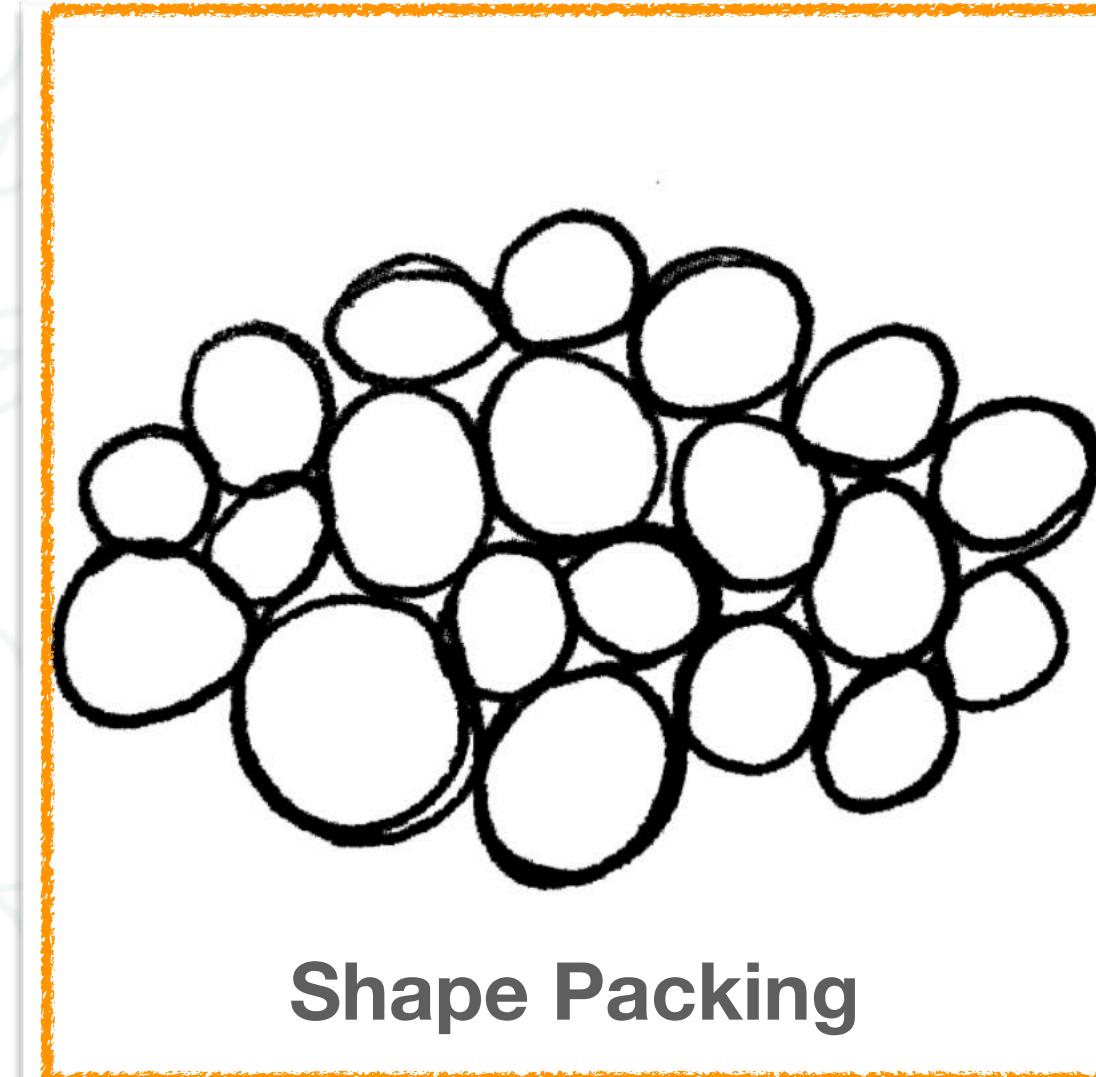
# Pattern Categorization



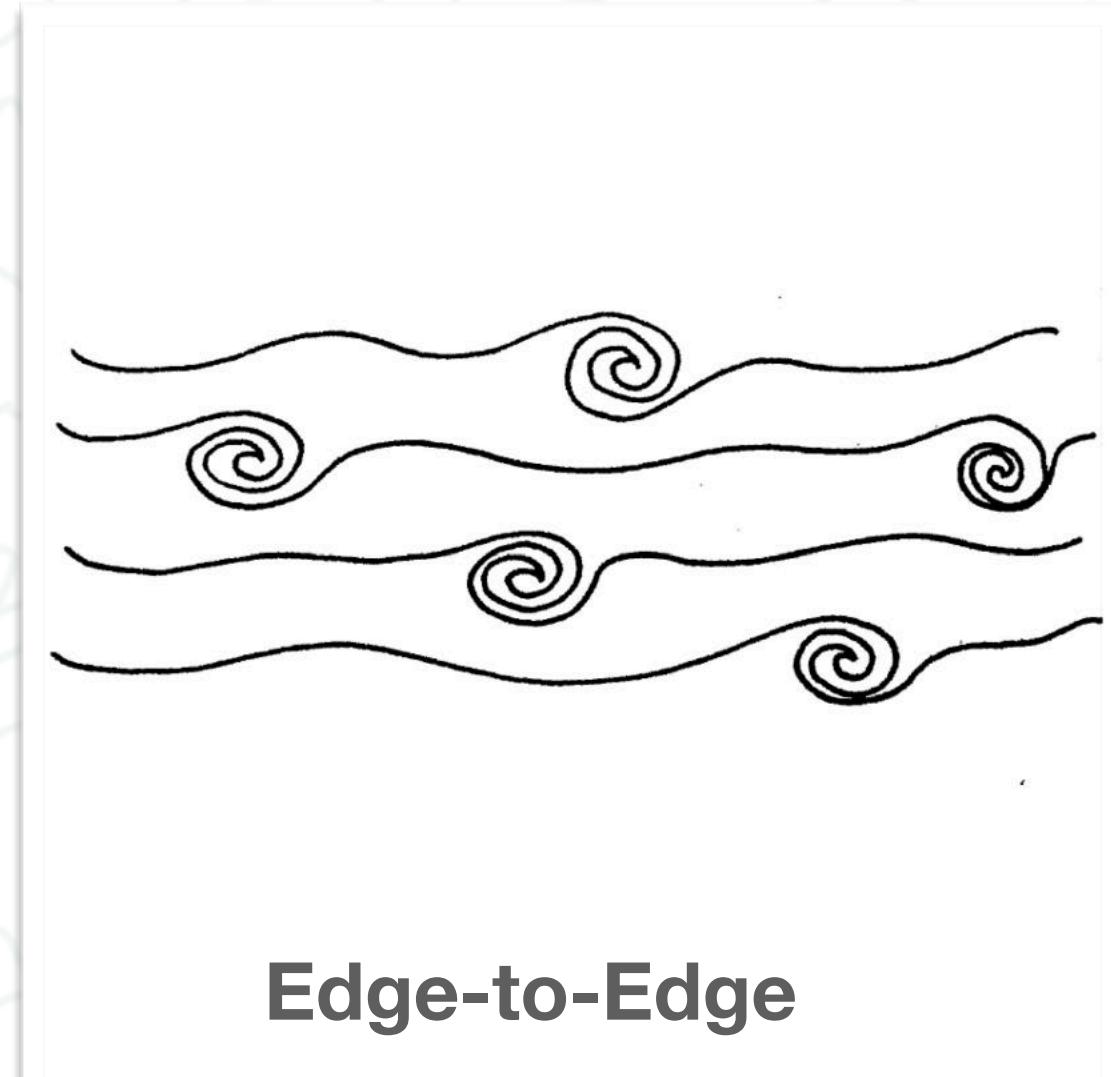
Emerging



Branching



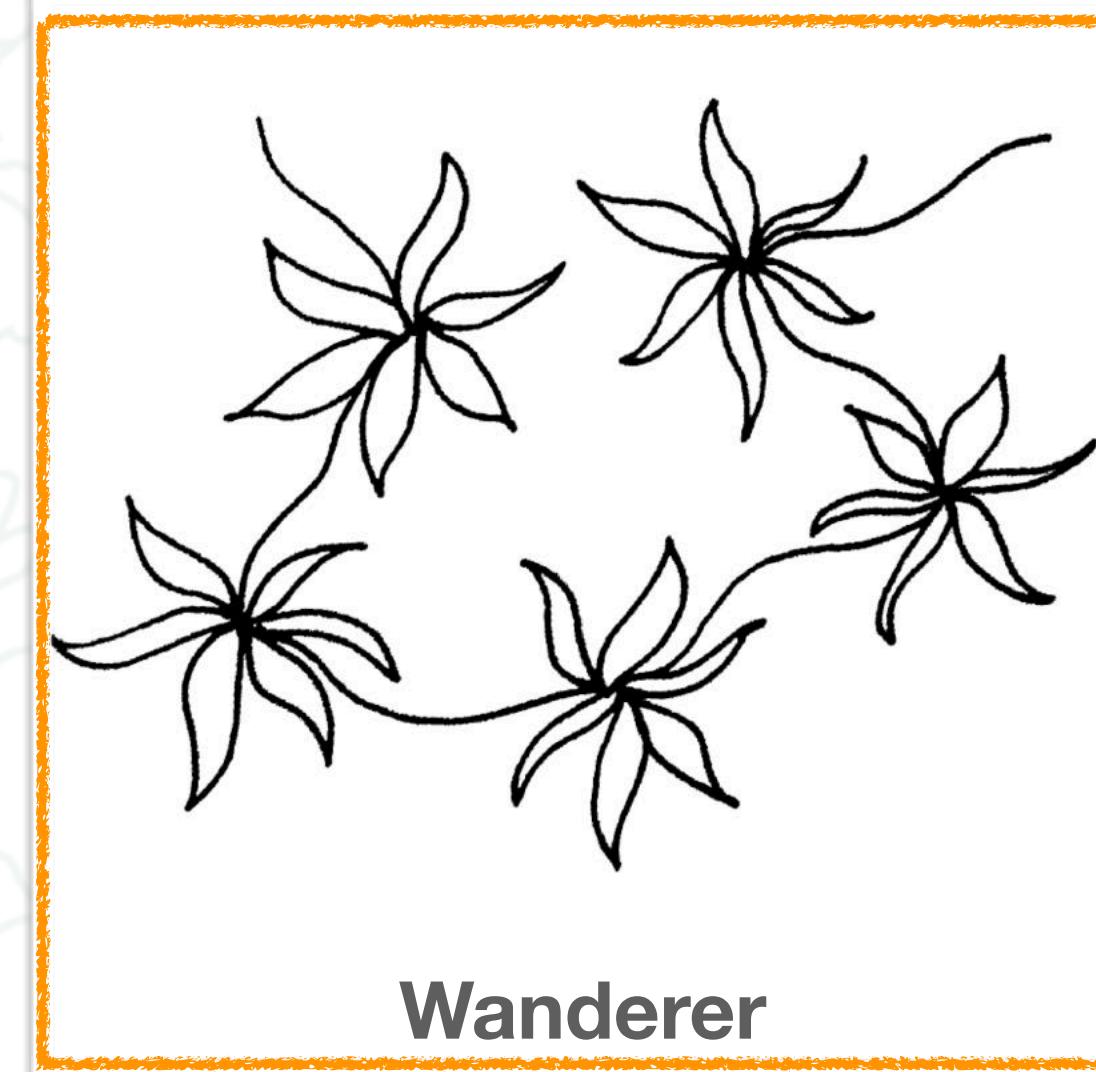
Shape Packing



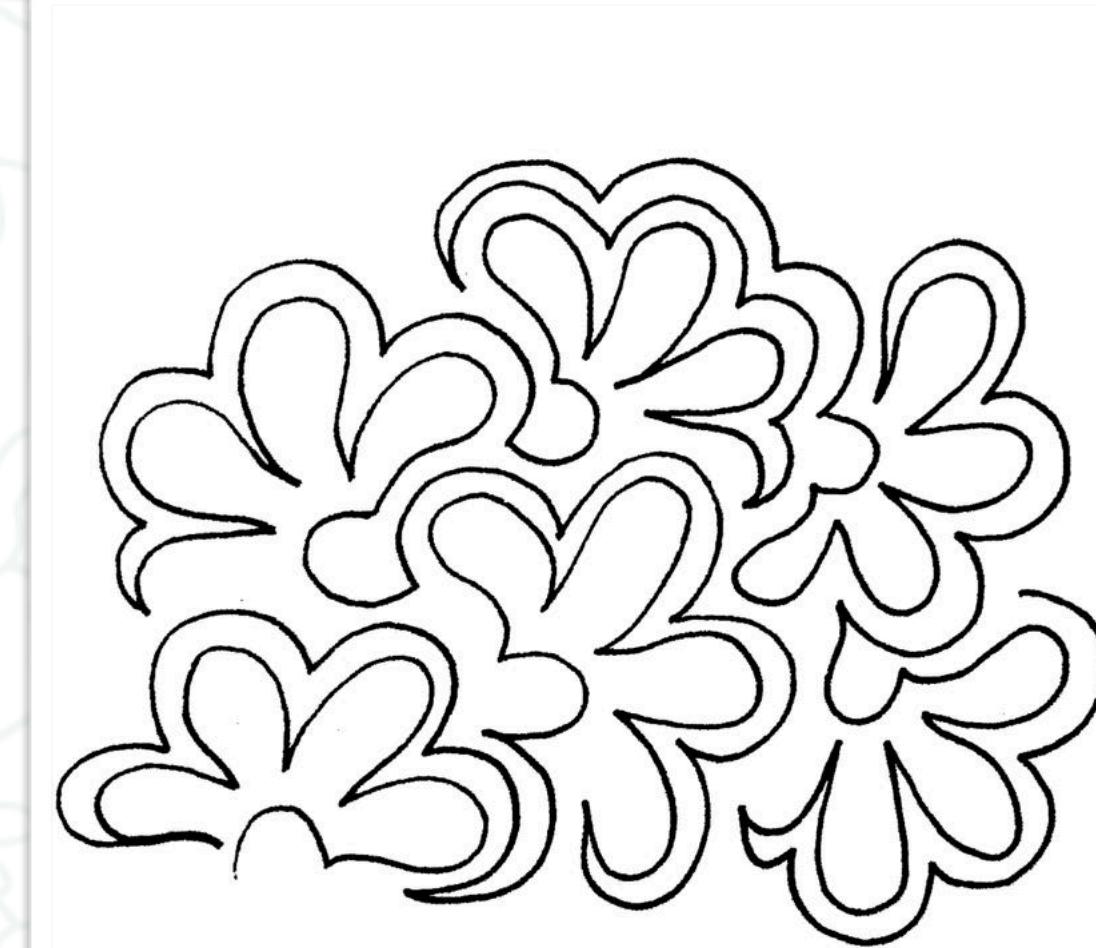
Edge-to-Edge



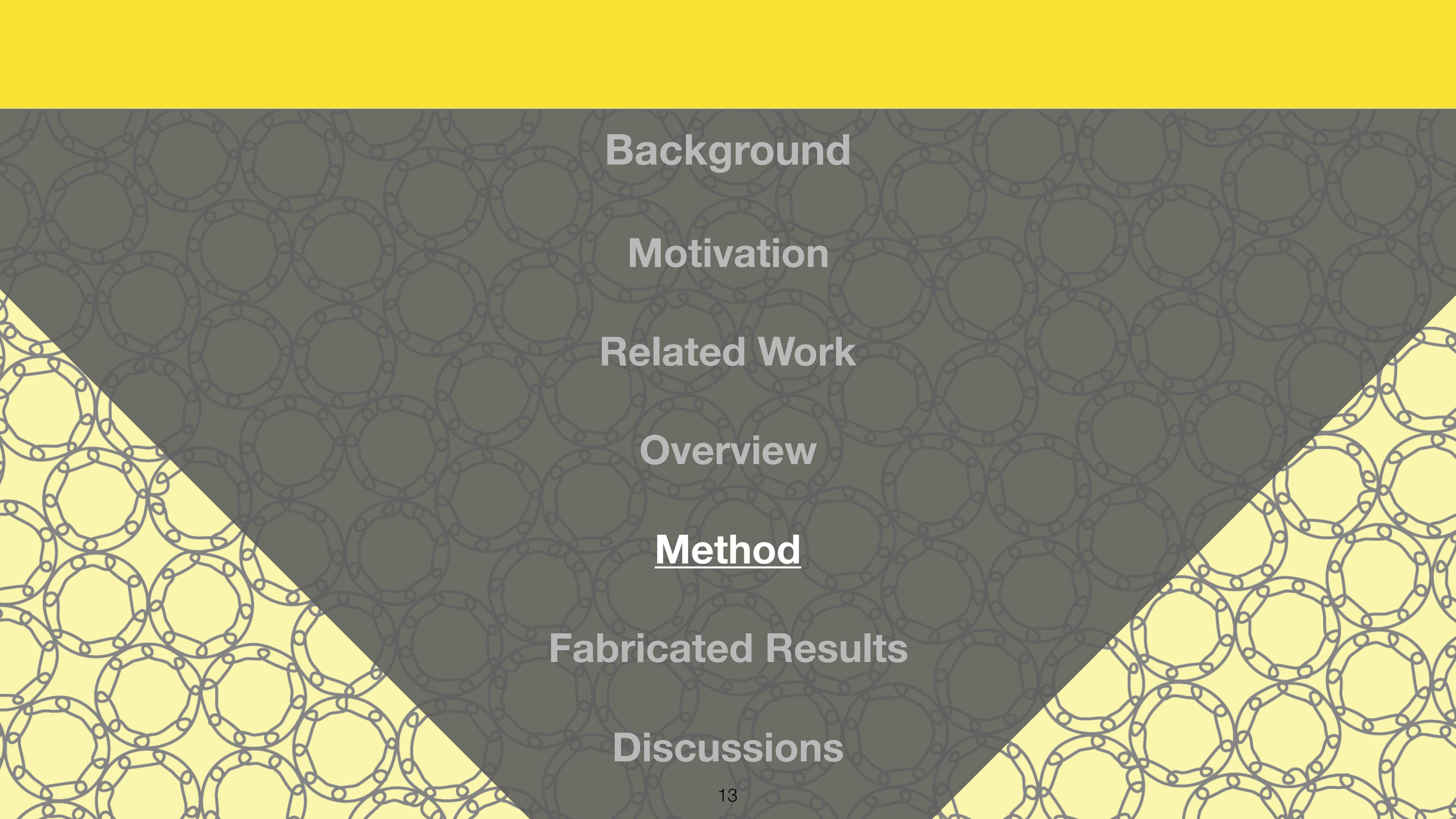
Climber



Wanderer



Back-and-forth Echoing



**Background**

**Motivation**

**Related Work**

**Overview**

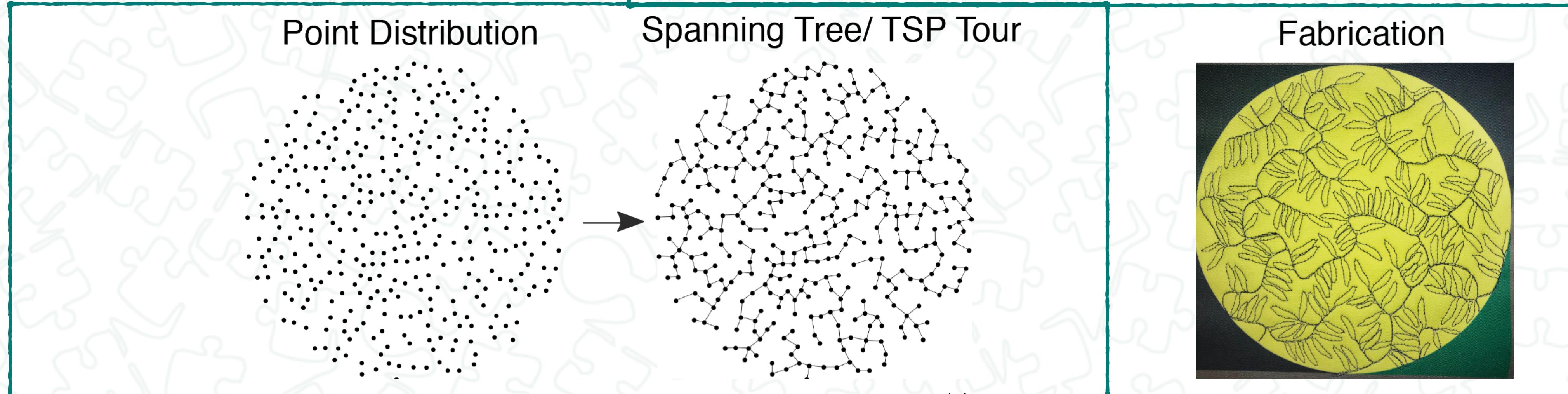
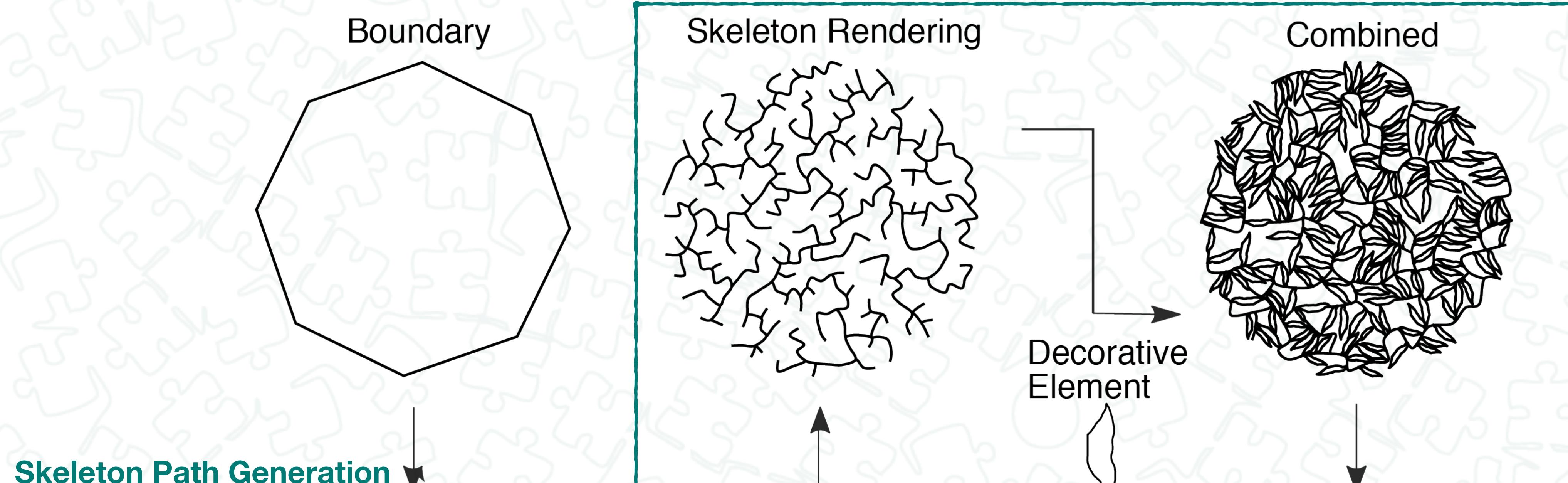
**Method**

**Fabricated Results**

**Discussions**

# Method

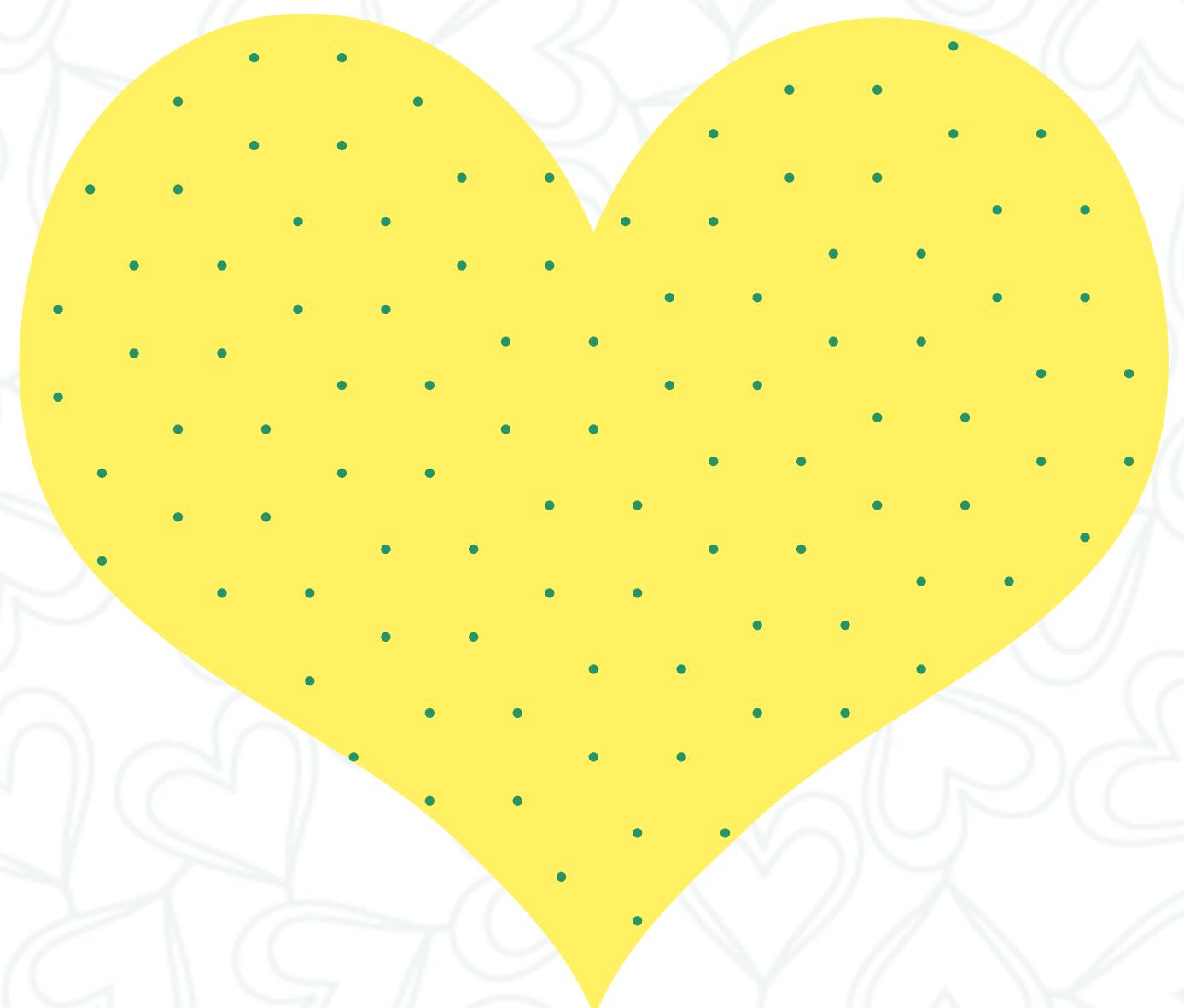
## Quilting Pattern Generation



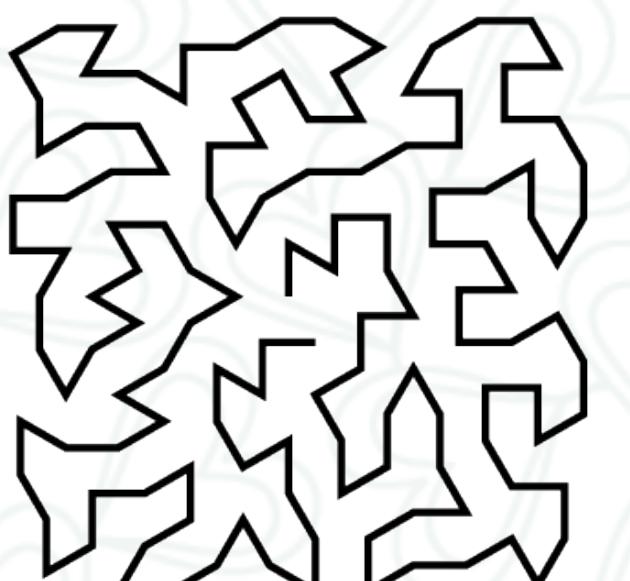
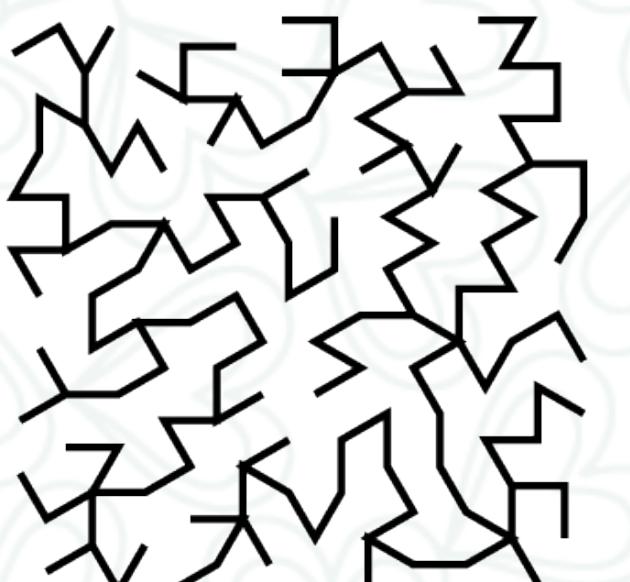
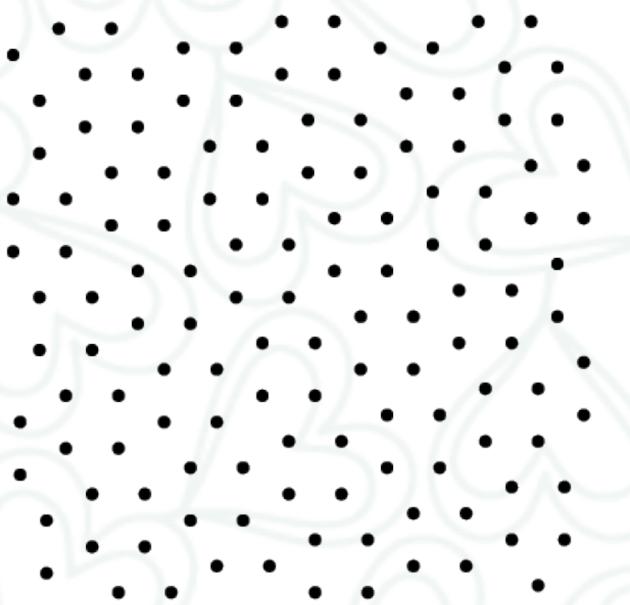
# Skeleton Path Generation

Point Distribution  
Path Conversion

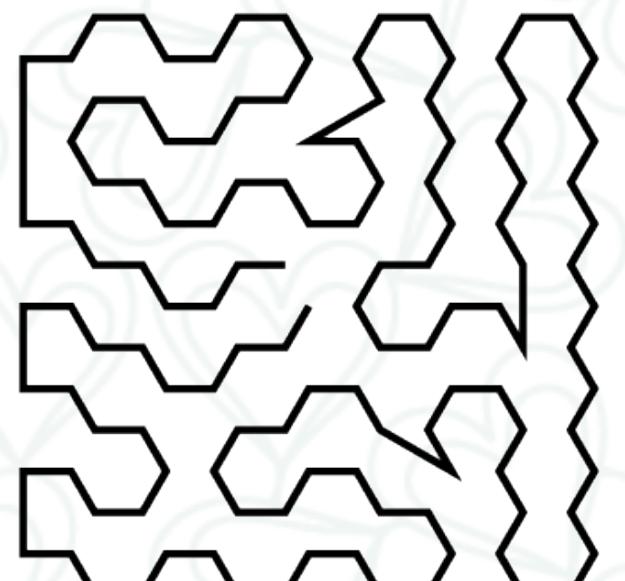
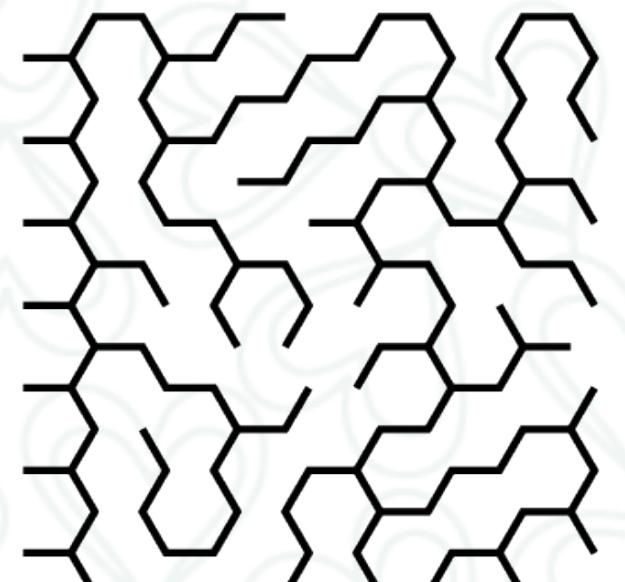
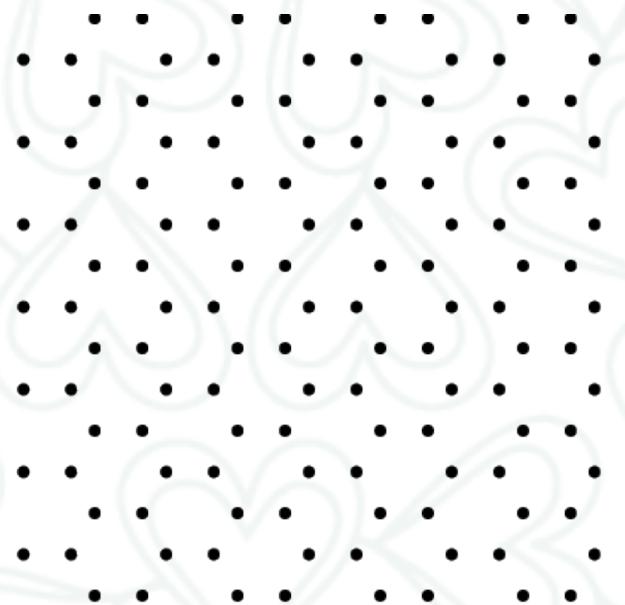
# Point Distribution



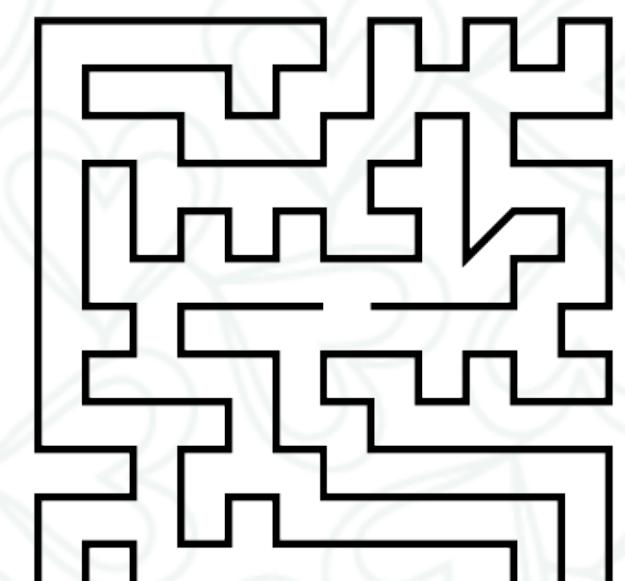
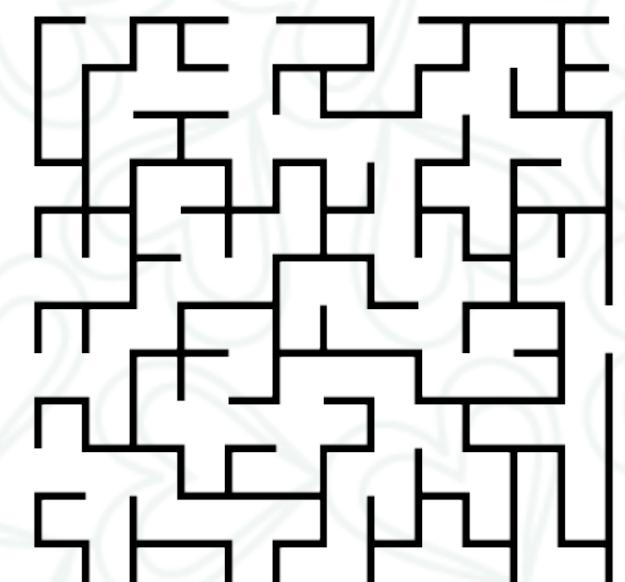
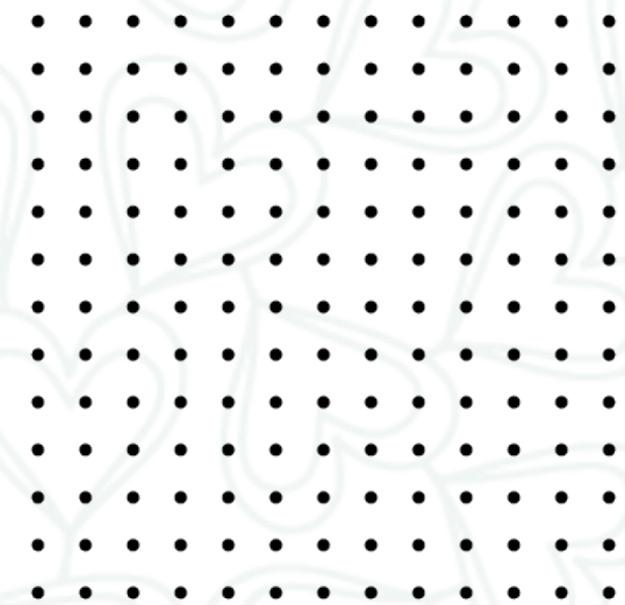
3.3.4.3.4  
tessellation



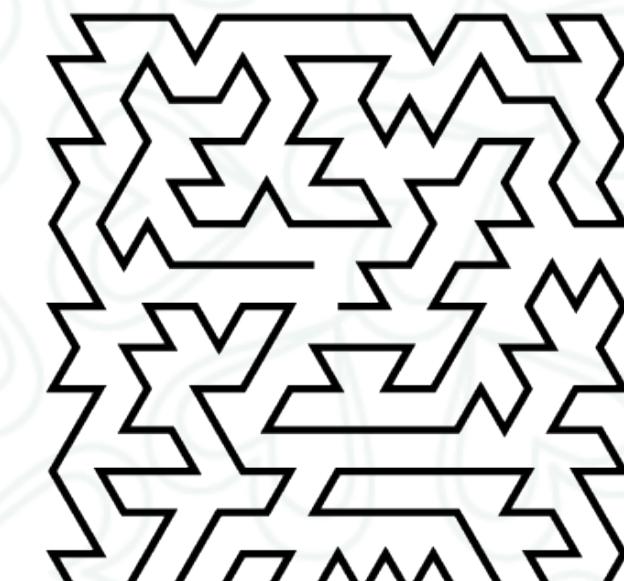
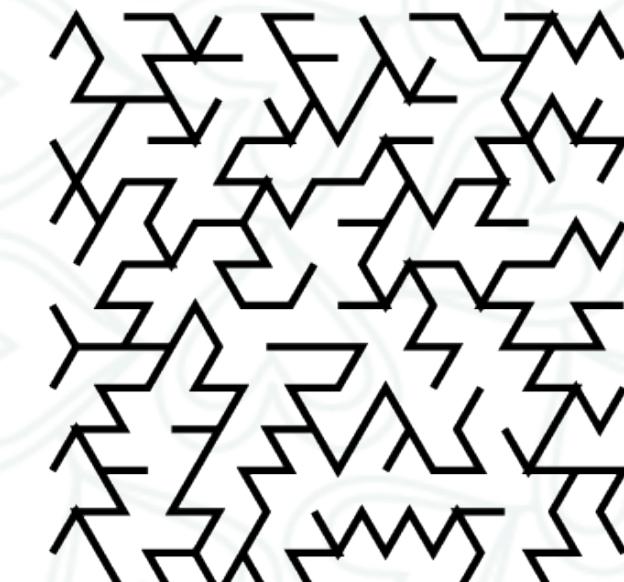
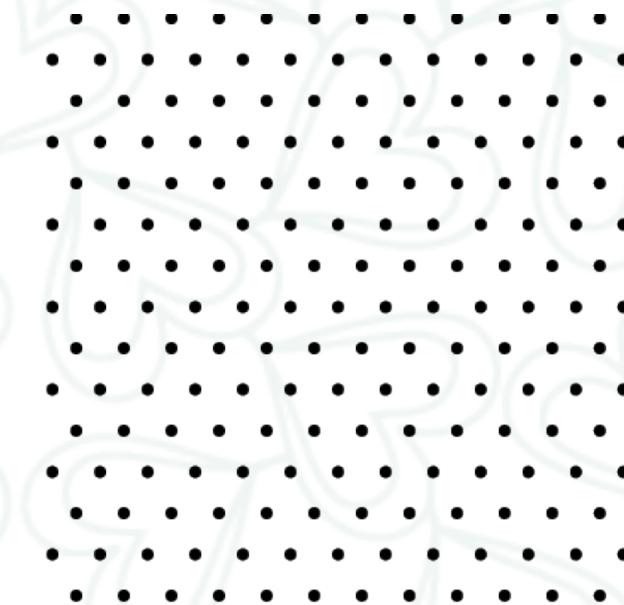
hexagonal  
tessellation



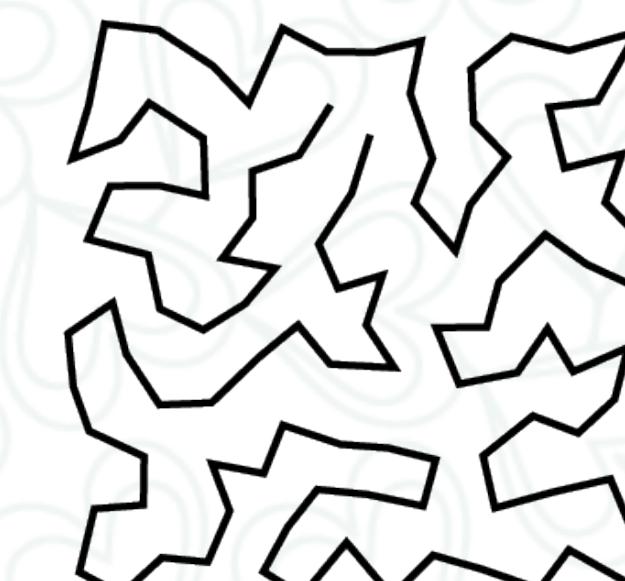
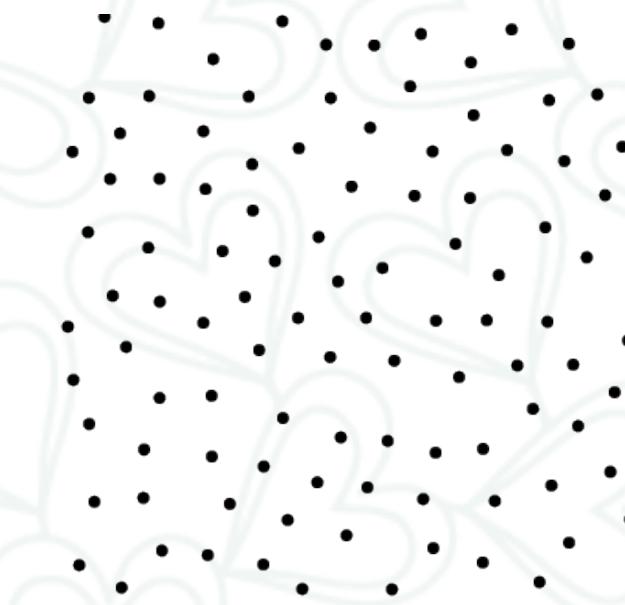
grid  
tessellation



triangle  
tessellation



Poisson-Disk  
sampling



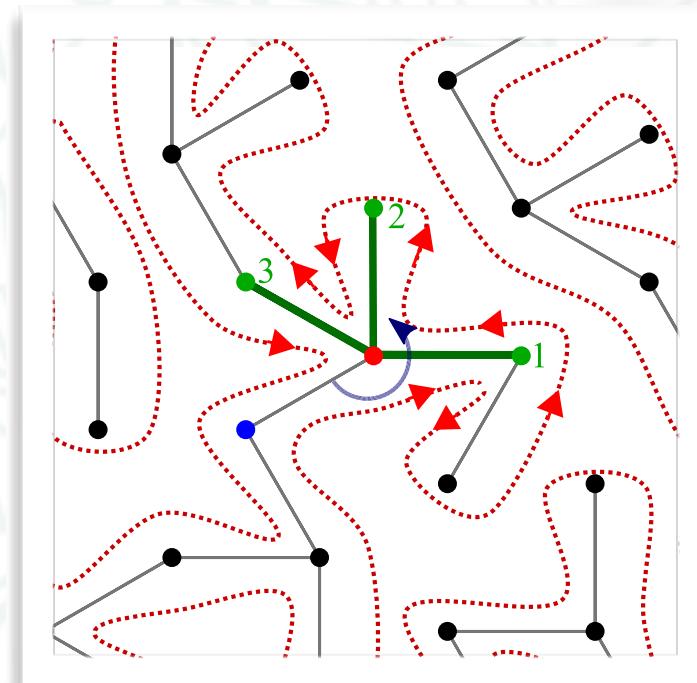
# Path Conversion



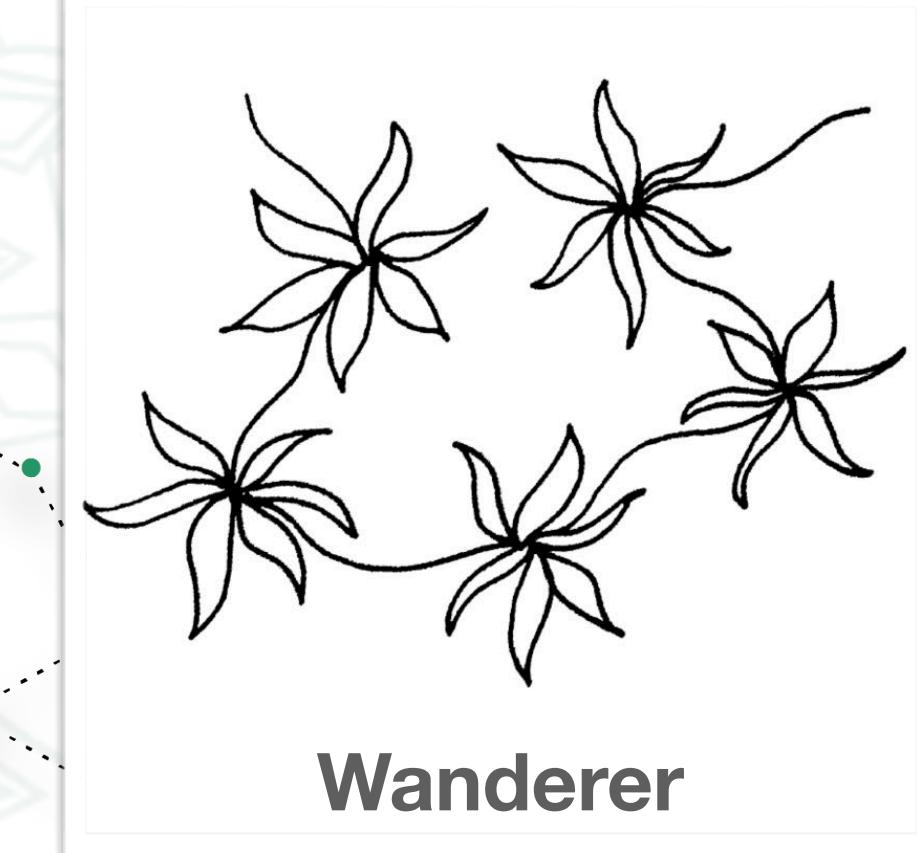
Branching



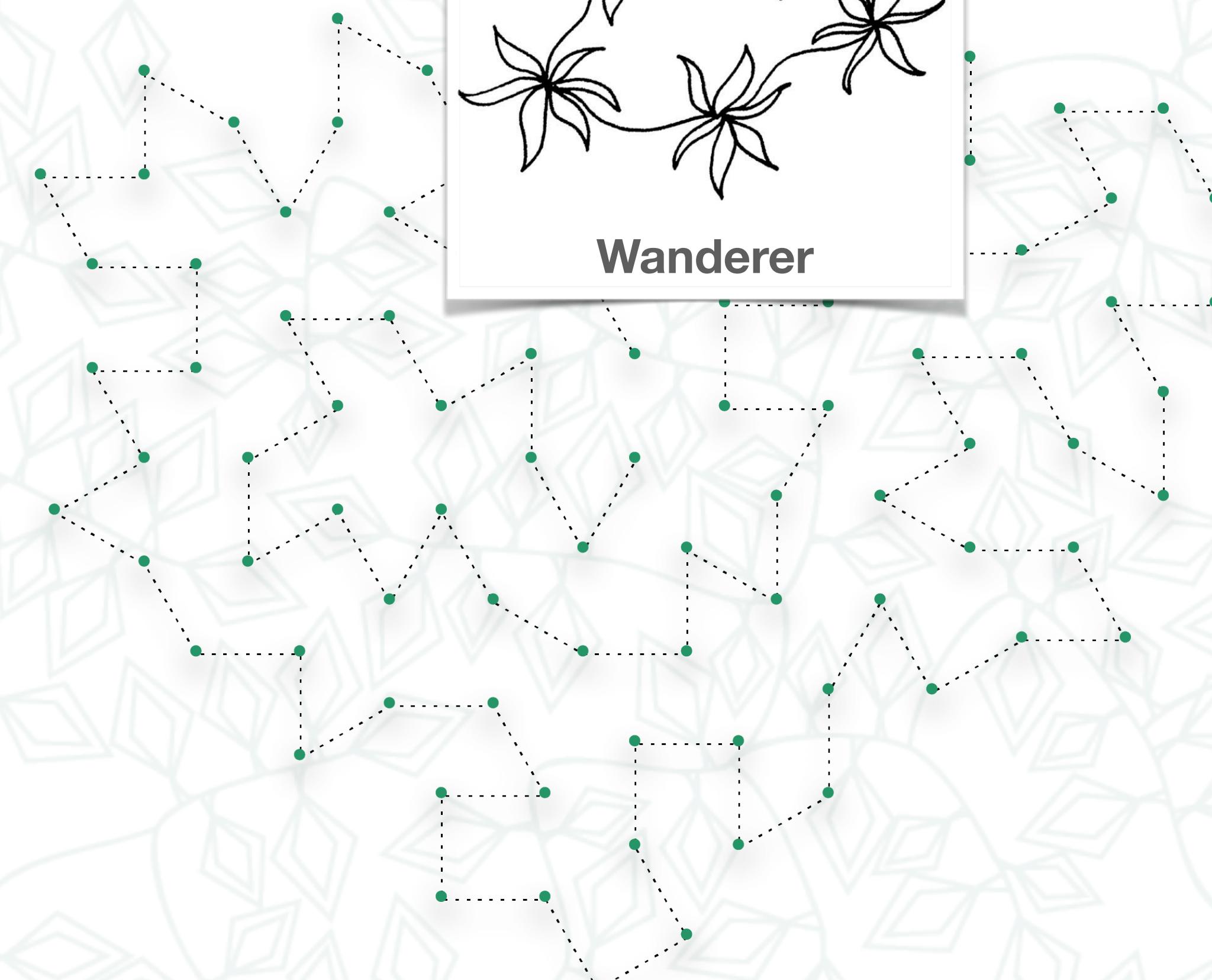
Shape Packing



Euclidean Minimum Spanning Tree (EMST)



Wanderer

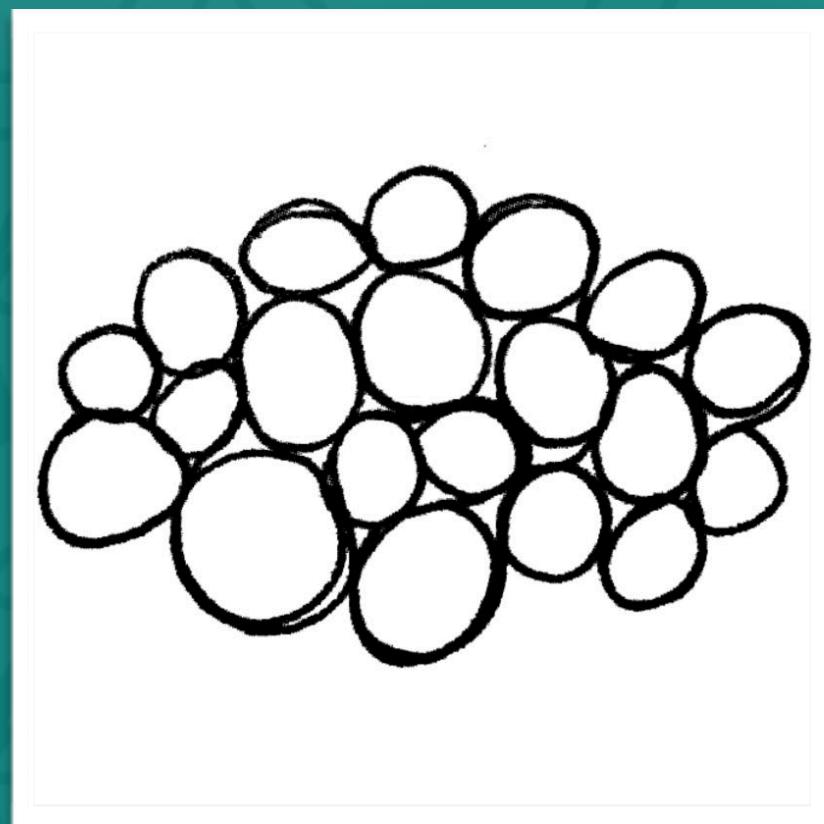


Travelling Salesman Path (TSP)

# Quilting Pattern Generation



Branching



Shape Packing



Wanderer

# Branching



Mimics the natural look of plants

Decorative elements are placed on  
alternating sides of the stitching path

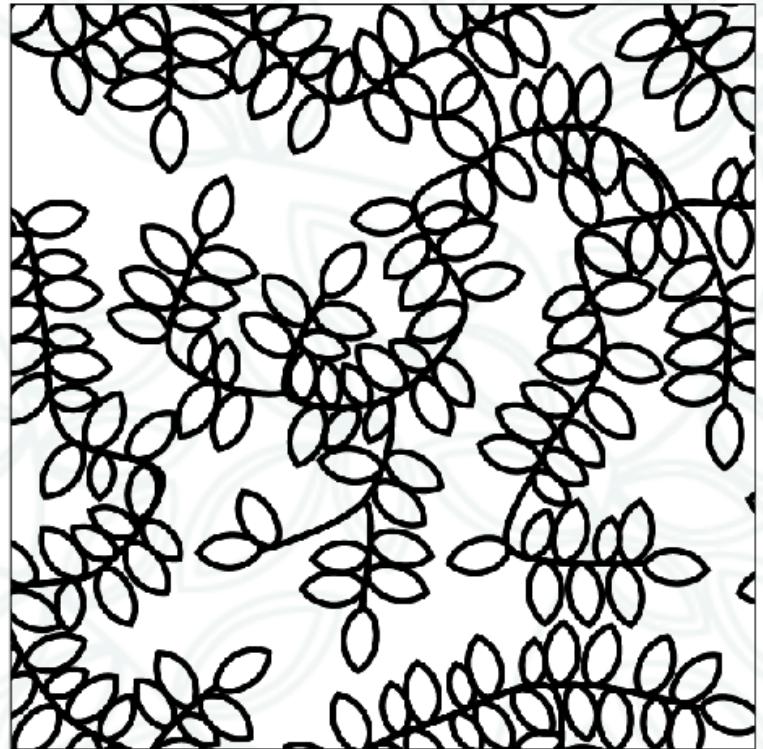
Point Distribution: Poisson-Disk  
Path Conversion: Spanning Tree



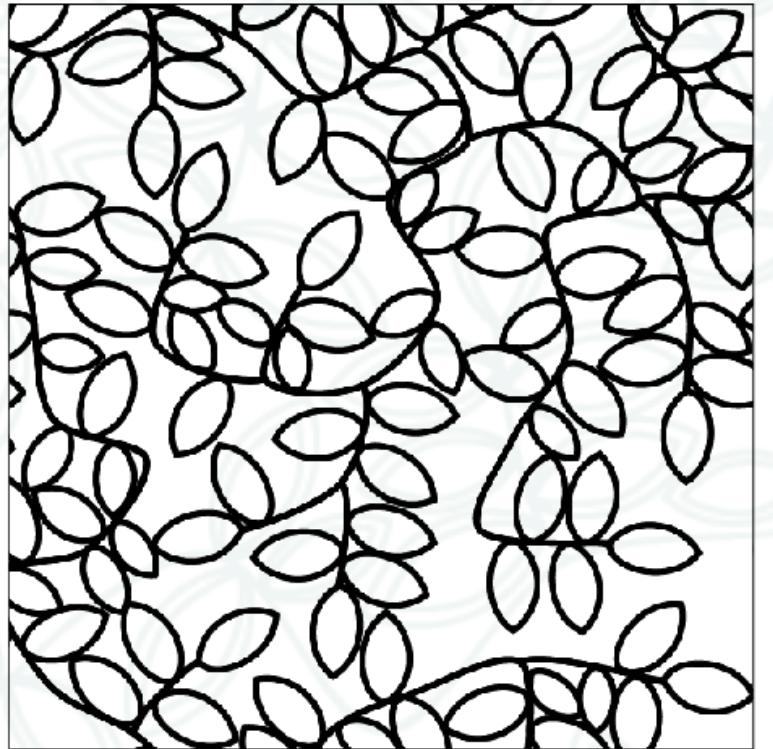
# Branching

## Parameterization:

1. Point distribution distance  $d$



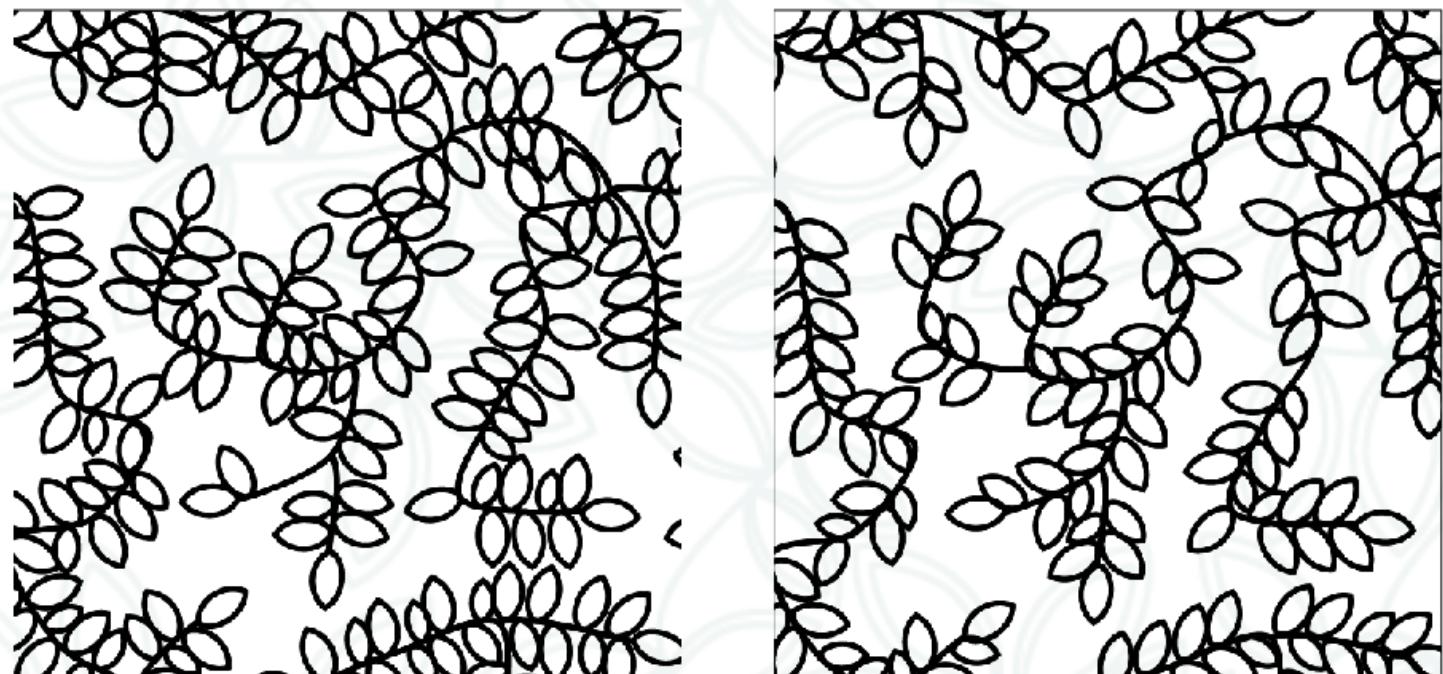
2. Decorative element scale  $s$



3. Gap length  $l$



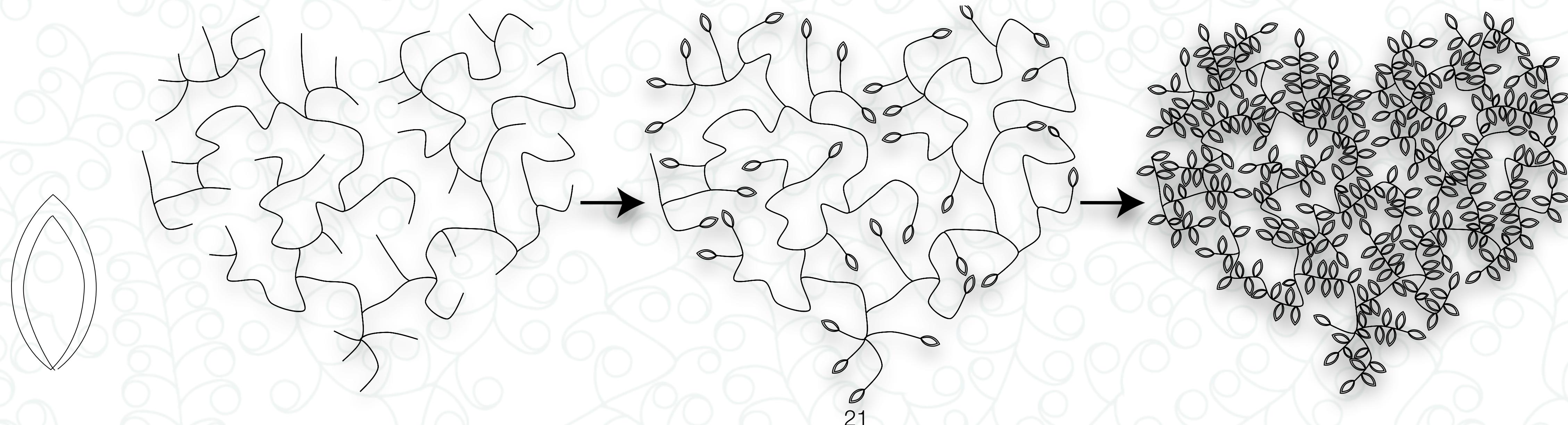
4. Angle  $\alpha$



# Branching

## Collision Detection/Resolution:

- Convex hull tests between the candidate and placed "leaves" + skeleton
- Collision resolution by rotating/scaling "leaves." Discard if all failed



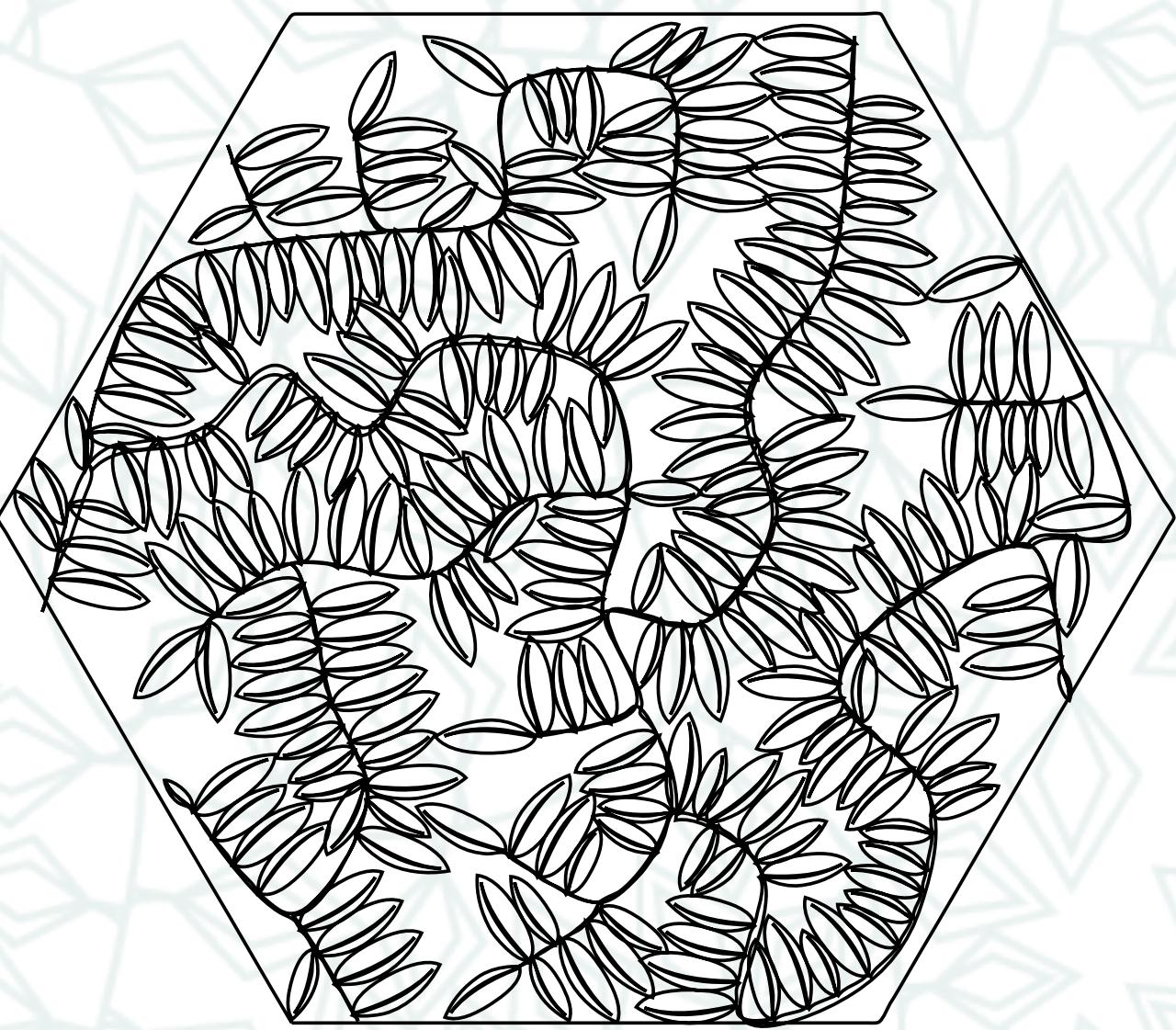
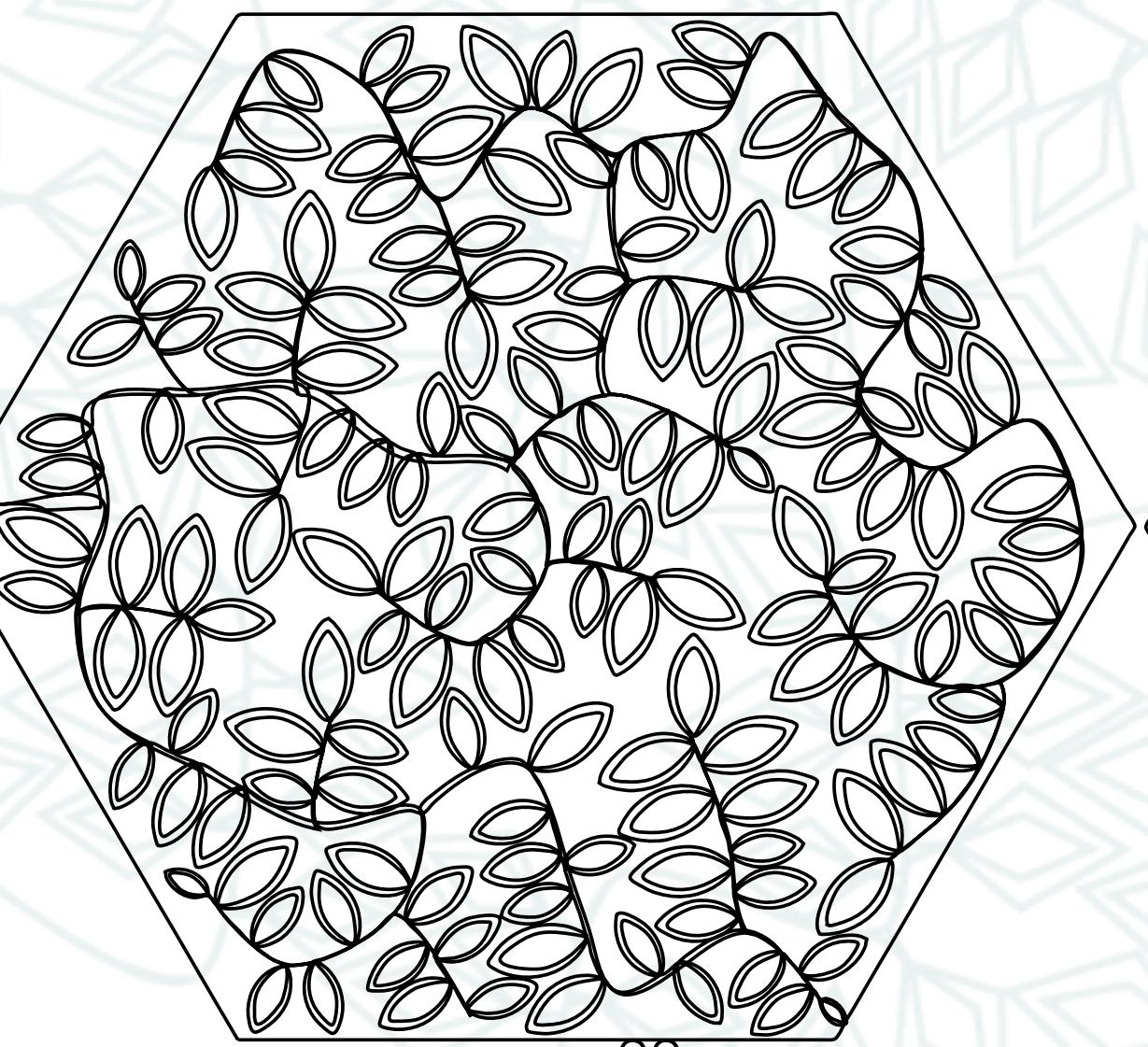
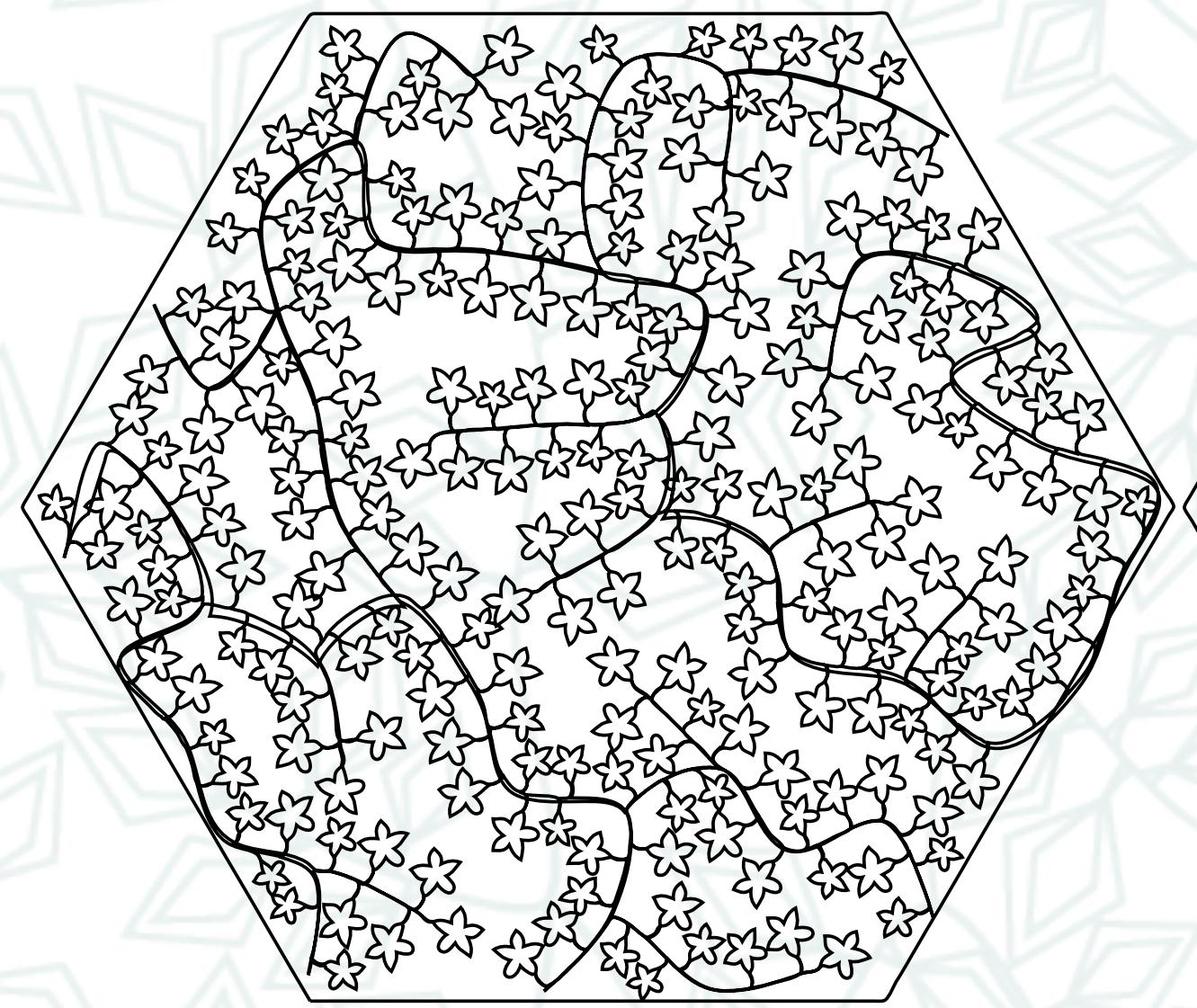
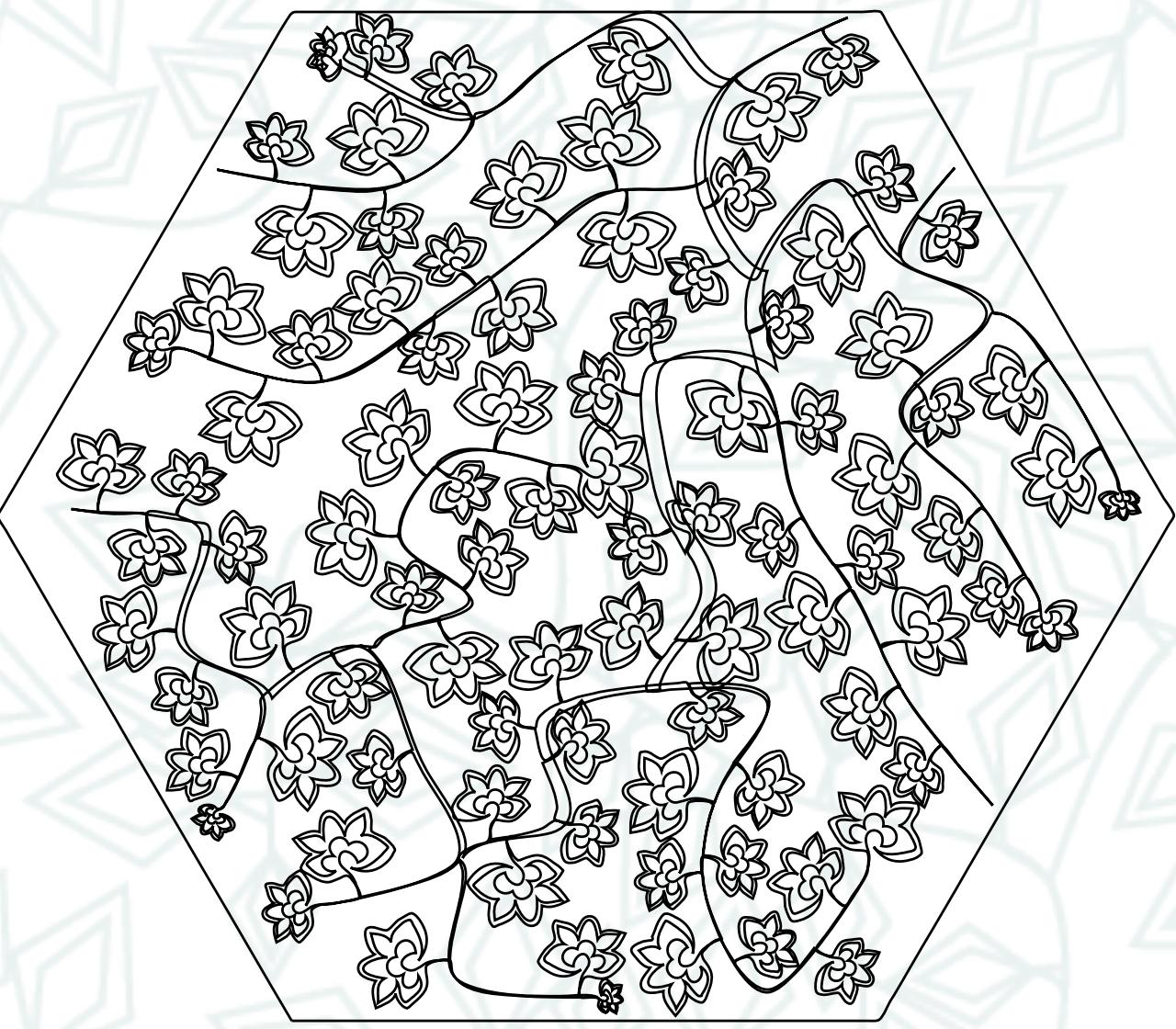
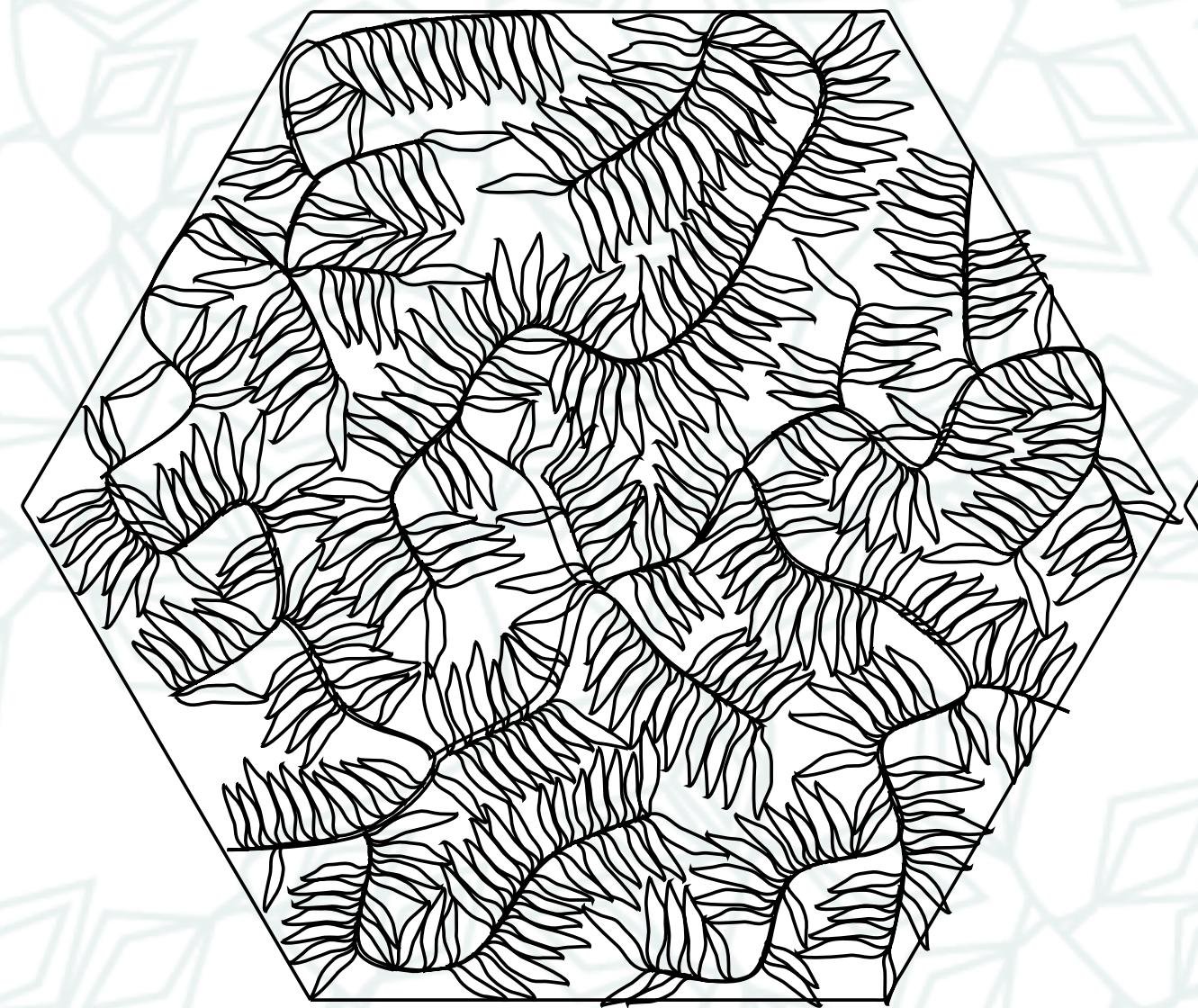
21

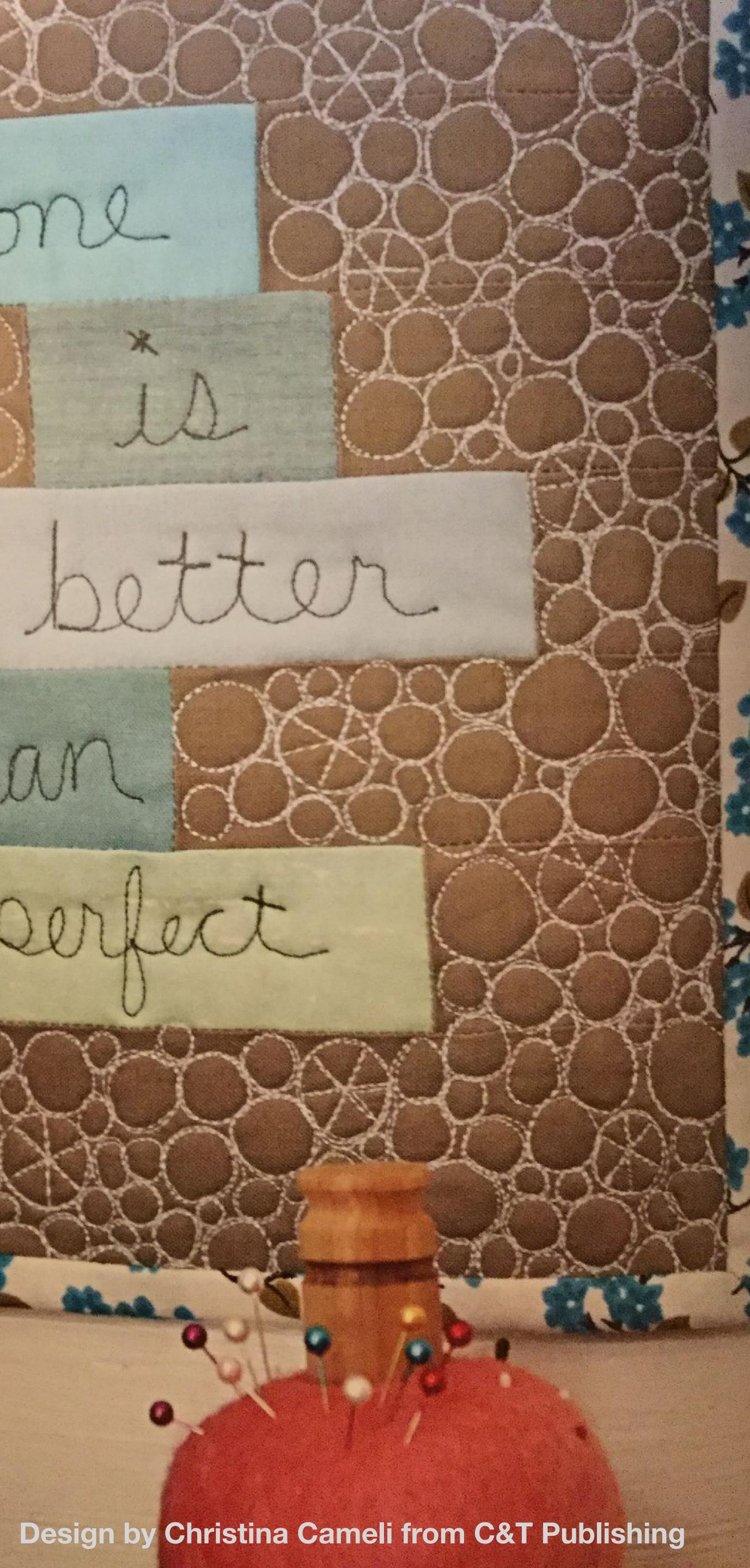
Interpolate skeleton path using  
Catmull-Rom splines

Attempts to place decorative elements  
at the tip of the branches.  
(Motivated by look of plants)

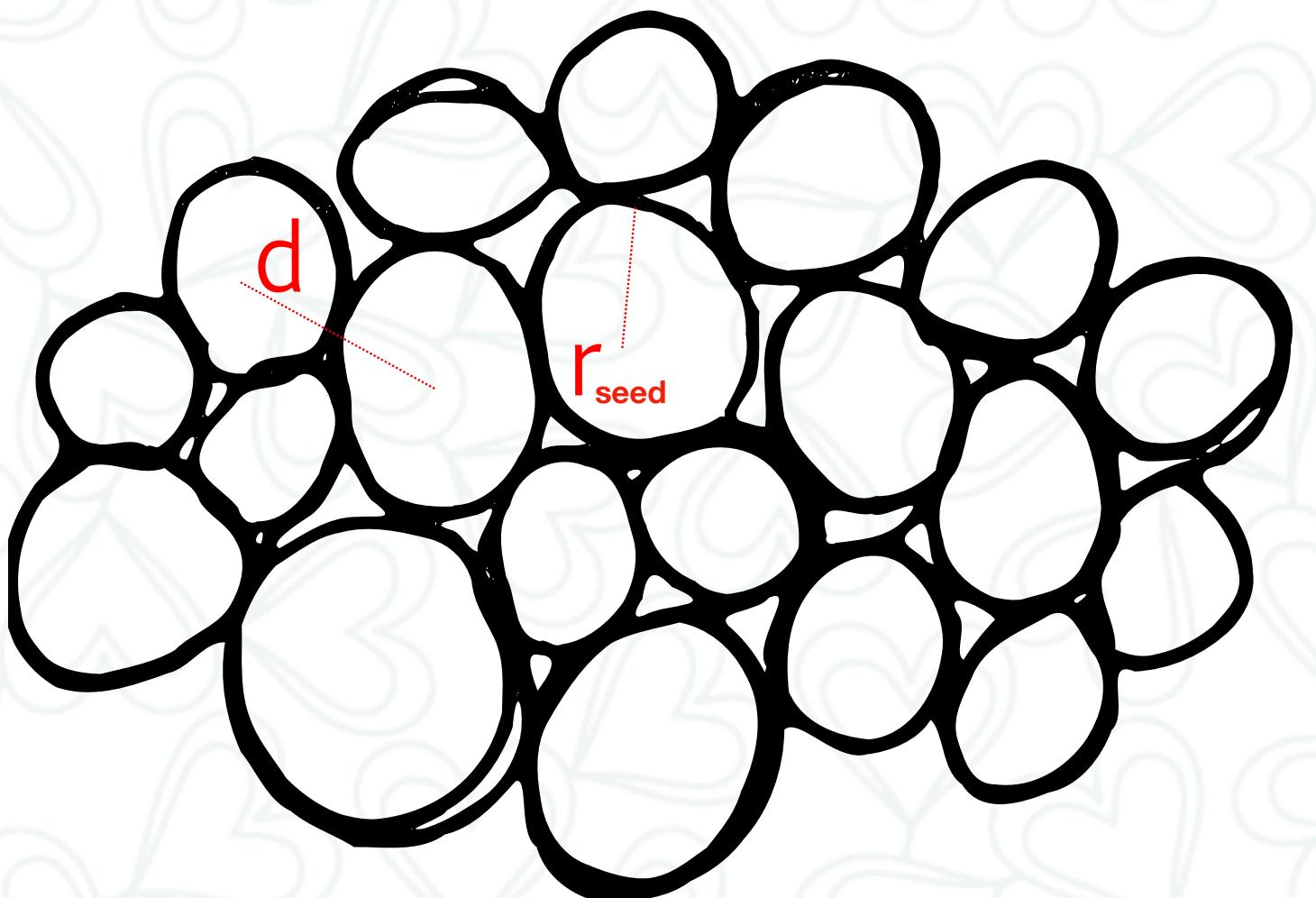
Attempts to place decorative elements along  
subdivided branches according user-specified  
direction

# Branching





# Shape Packing



**Point Distribution: Tessellations**

**Path Conversion: Spanning Tree**

Packs primitives as tightly as possible

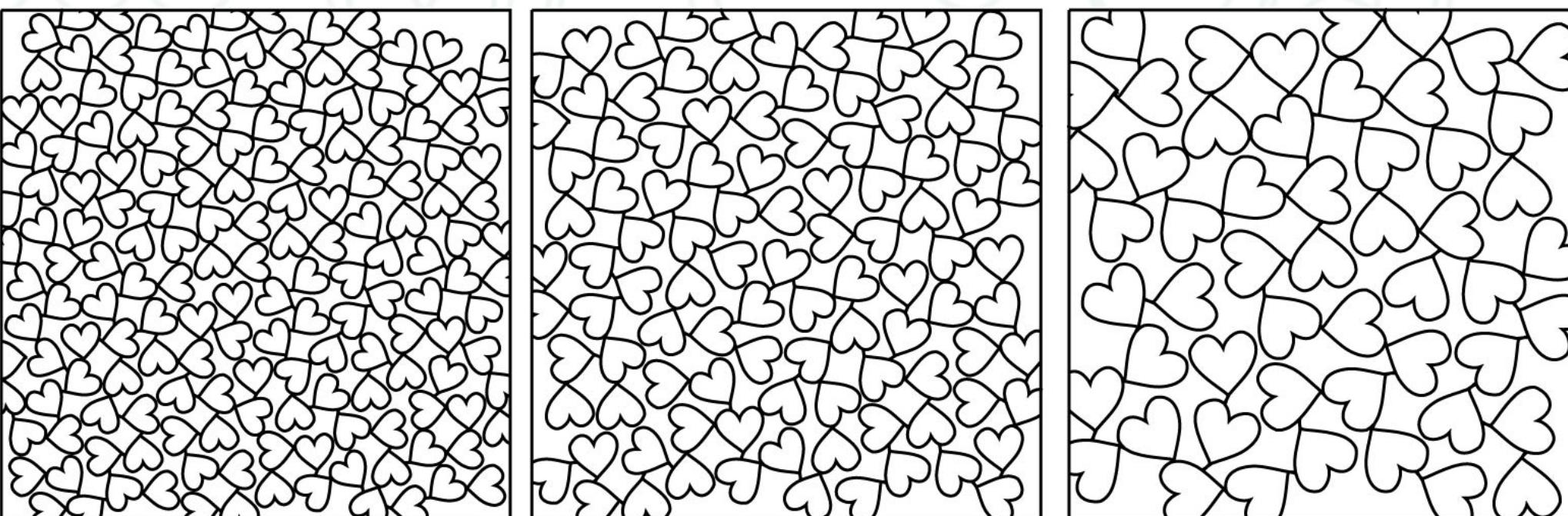
Various primitive sizes for random appearance

"Pebbling" = Shape packing of circles

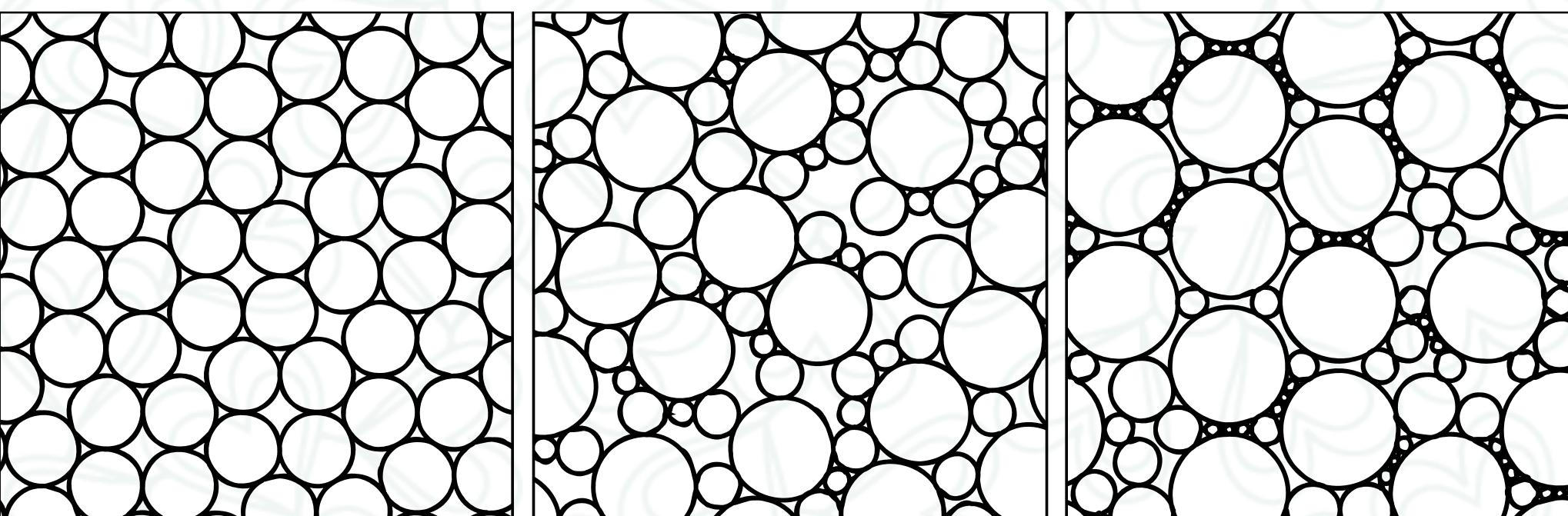
# Shape Packing

## Parameterization:

1. Point distribution distance  $d$



2. Initial size  $r_{seed}$

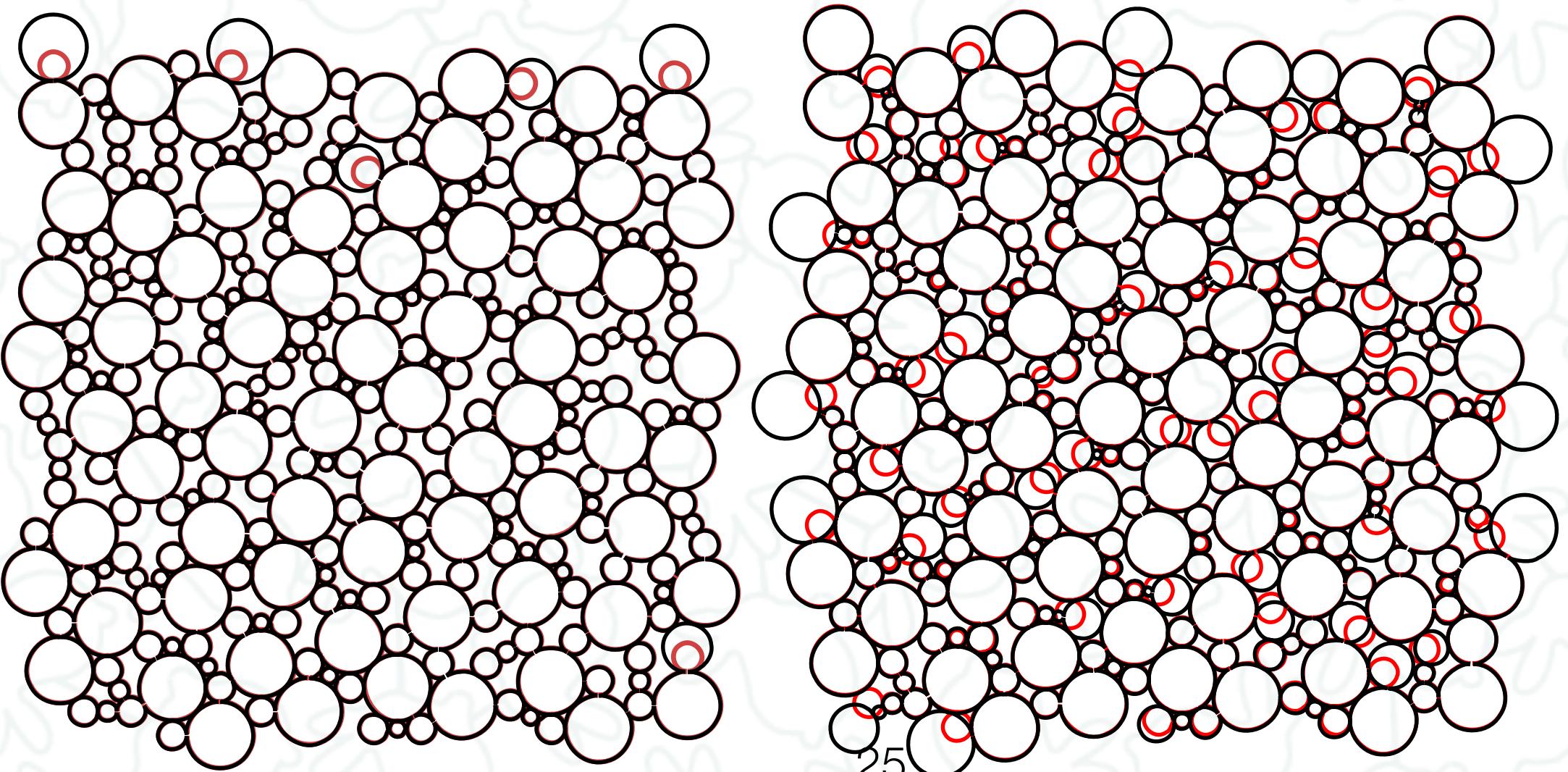
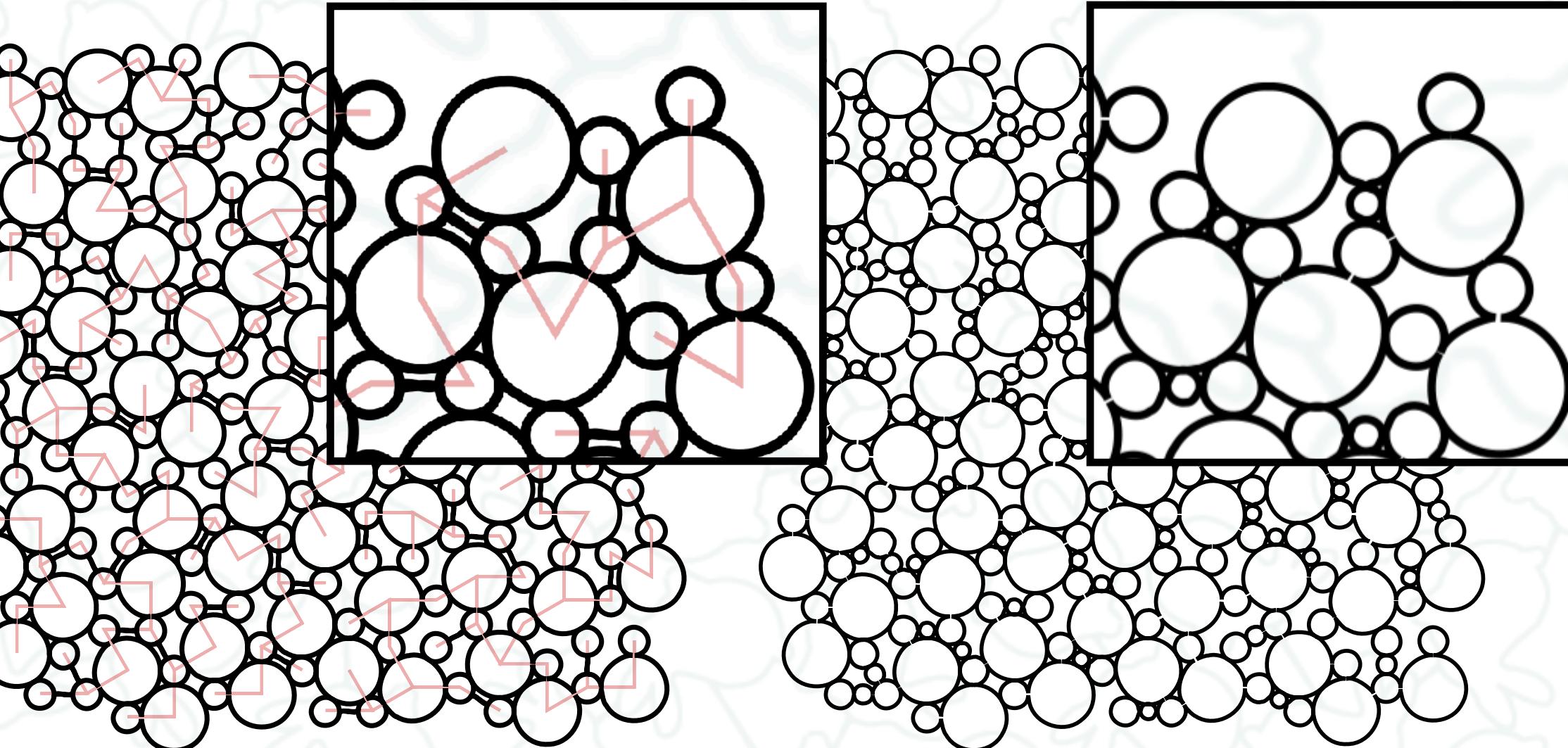
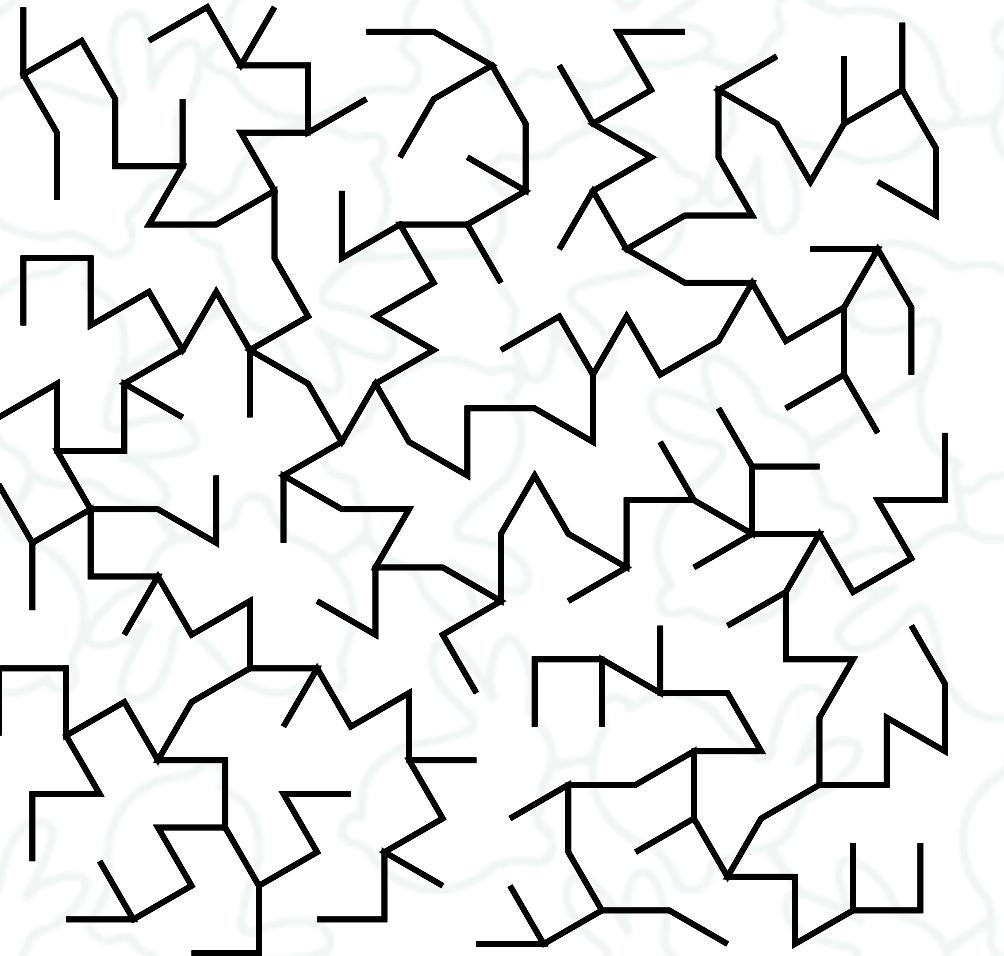


$$r_{seed} = 0.5d$$

$$r_{seed} = 0.4d$$

$$r_{seed} = 0.3d$$

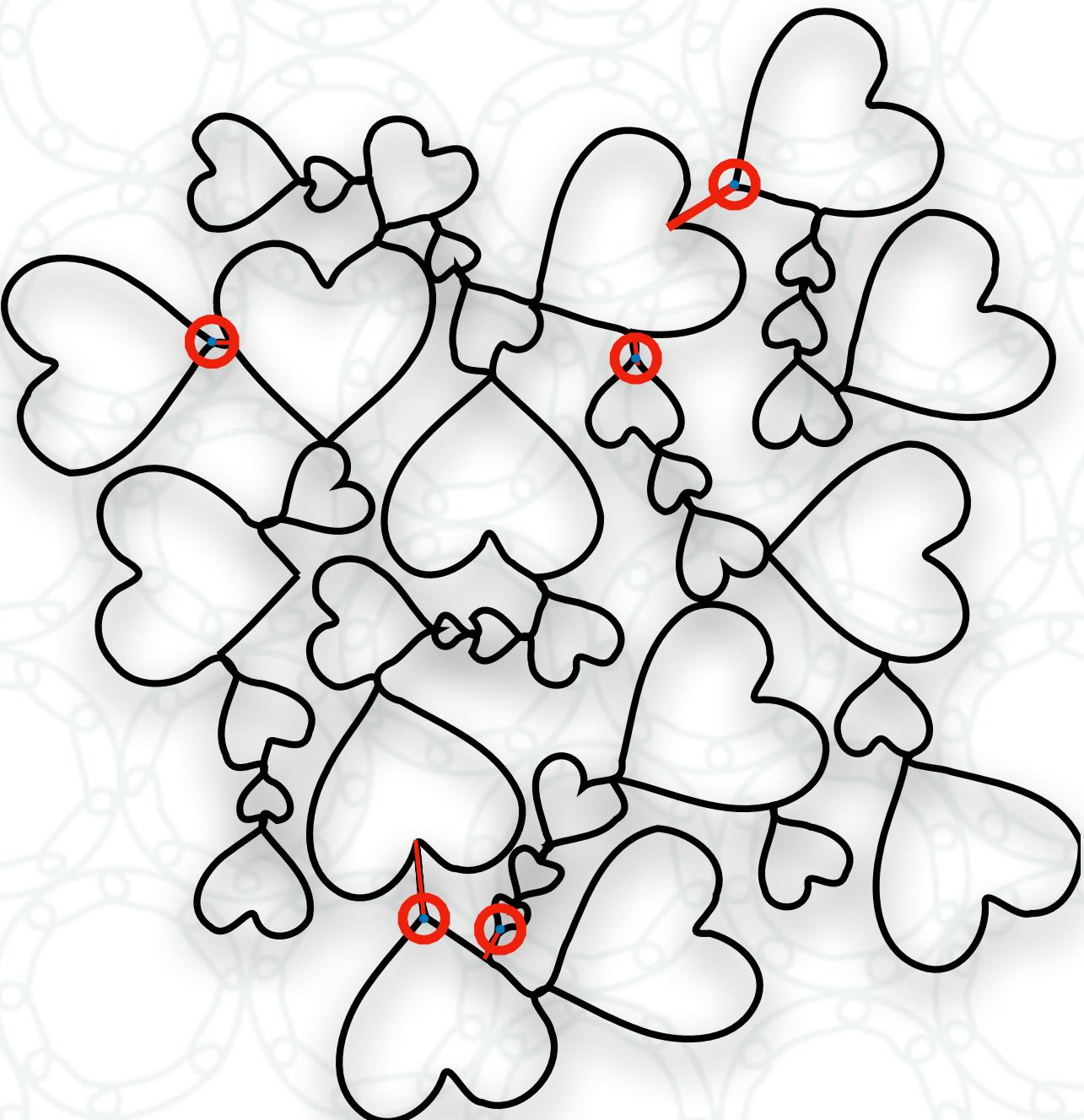
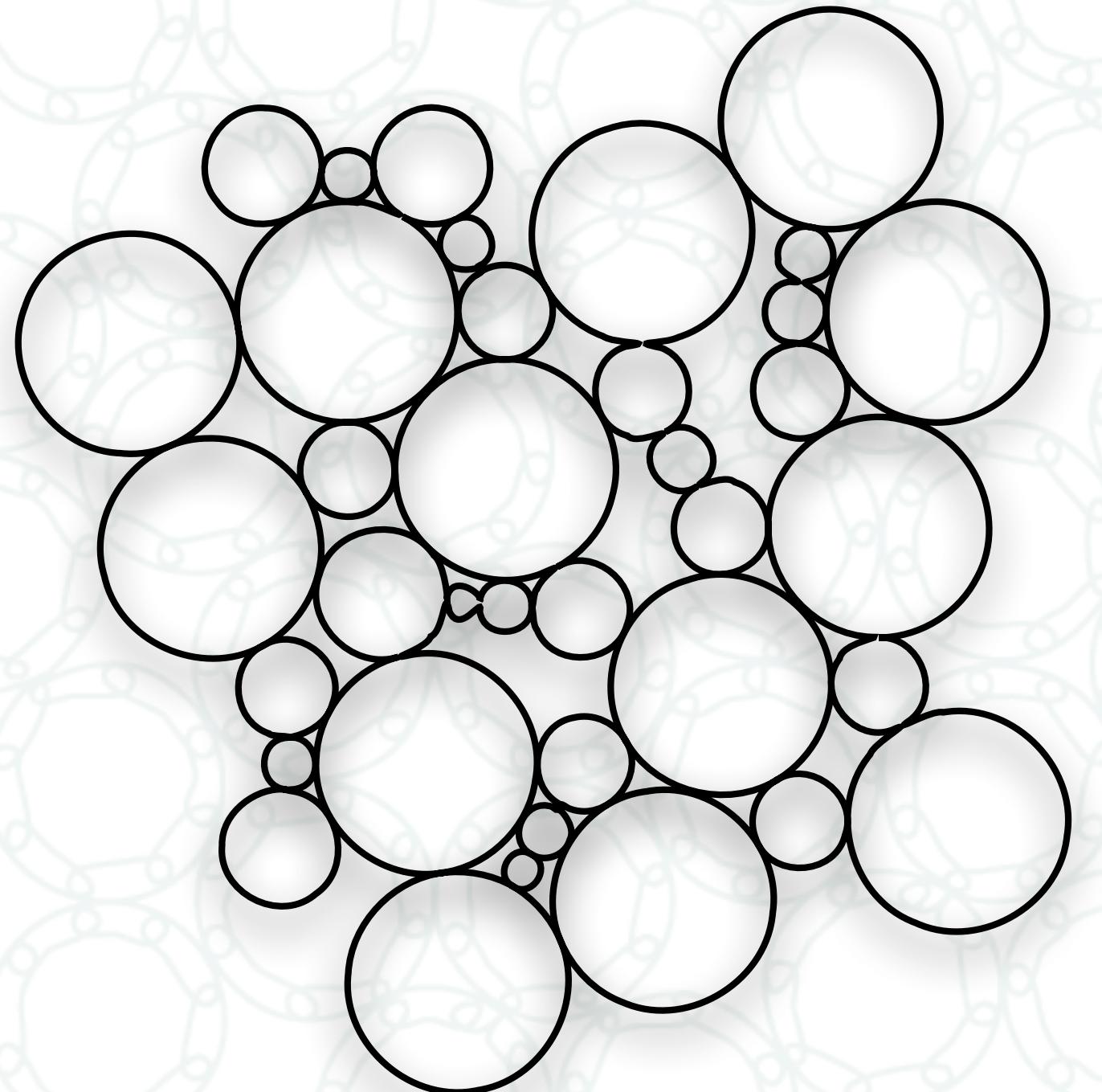
# Shape Packing



# Shape Packing



Connection points not on  
one of the original shape

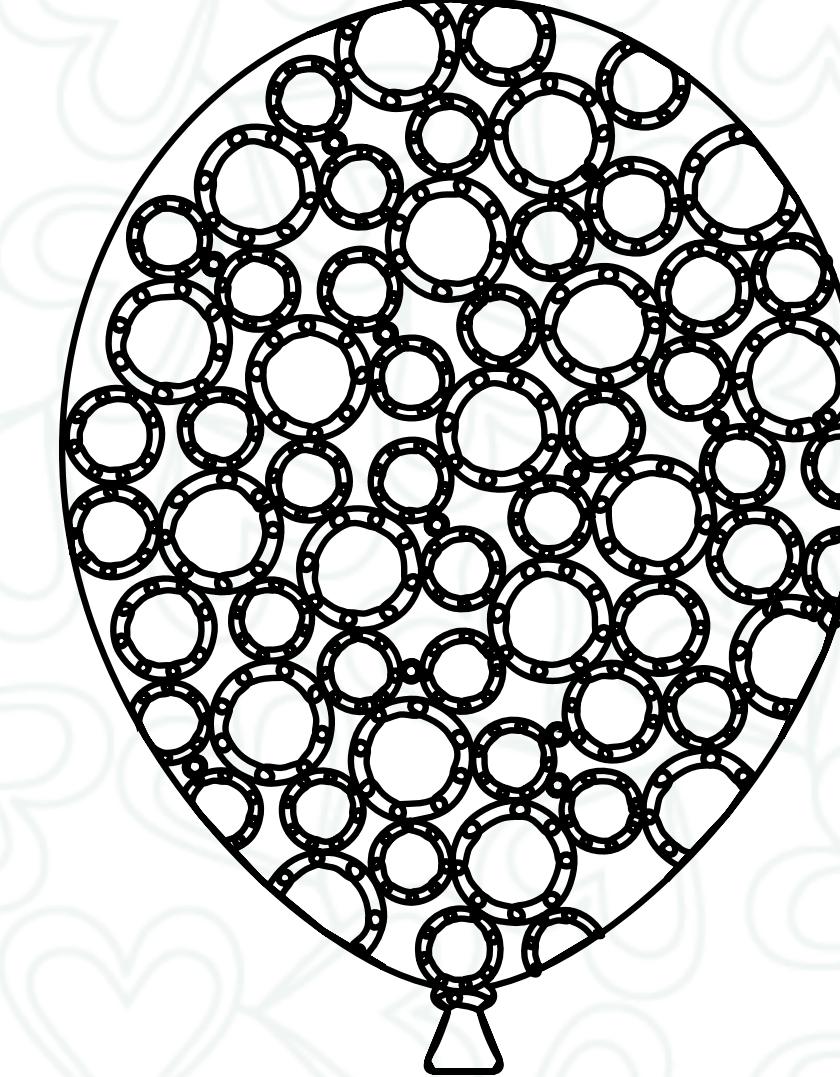
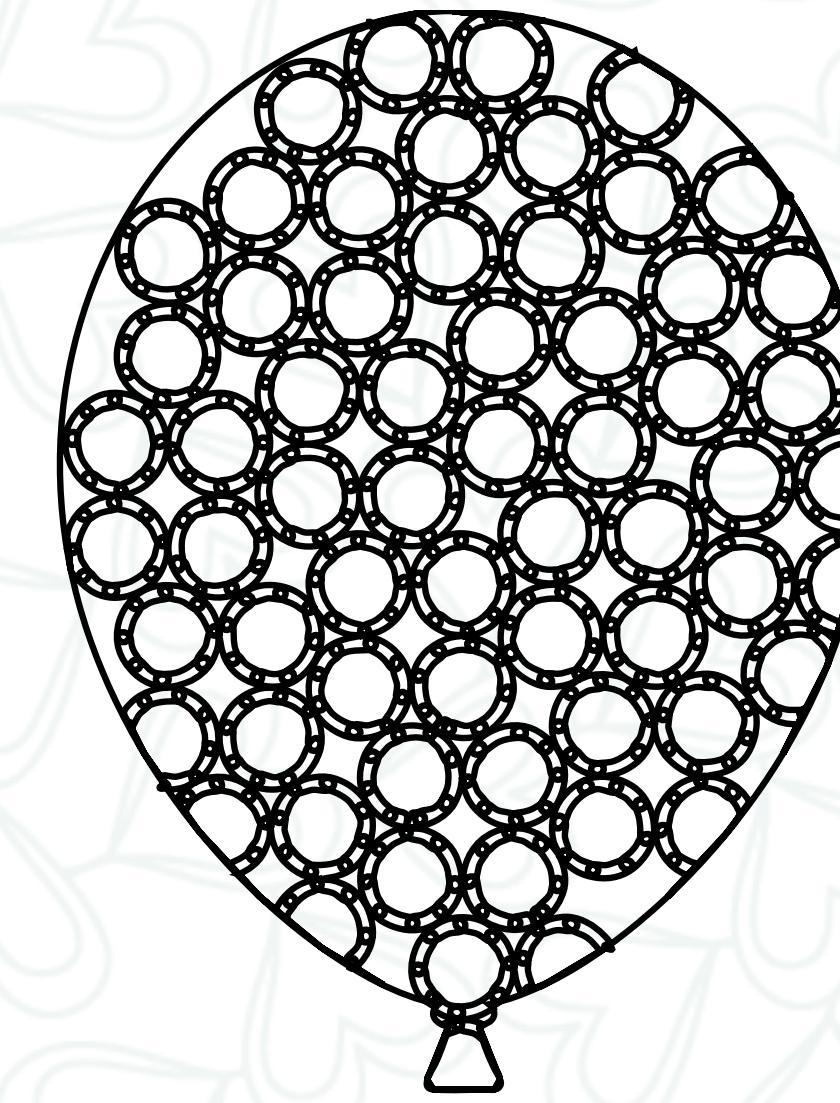
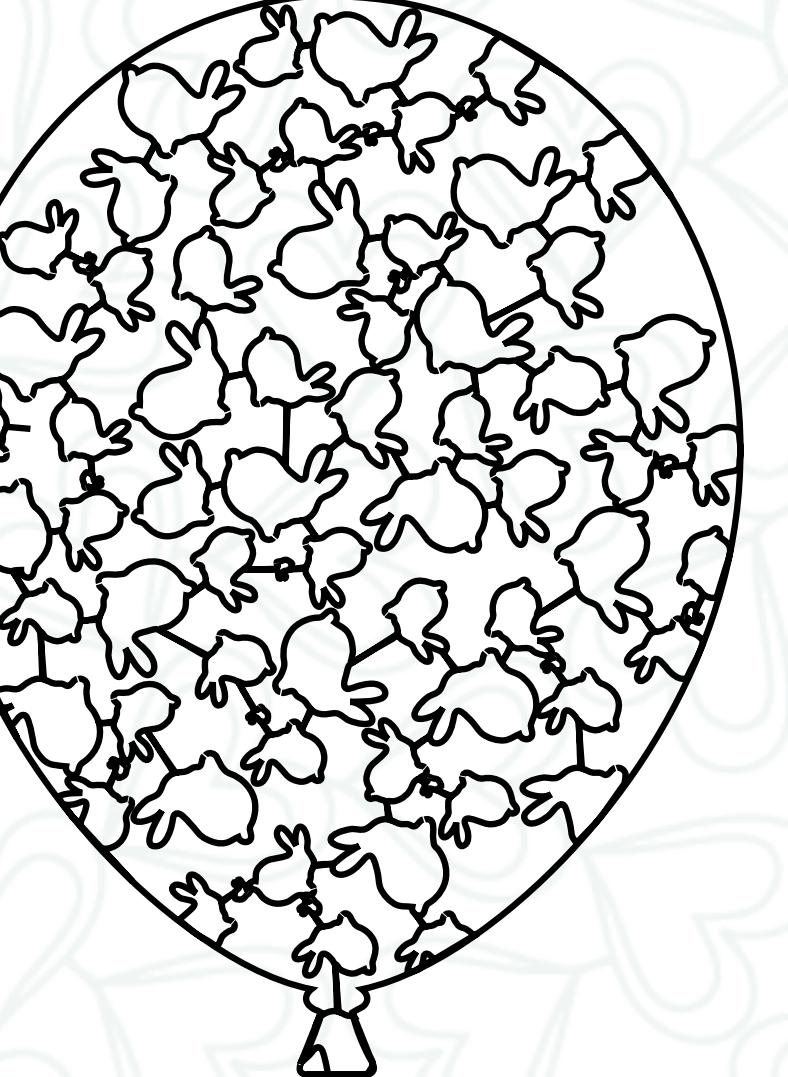
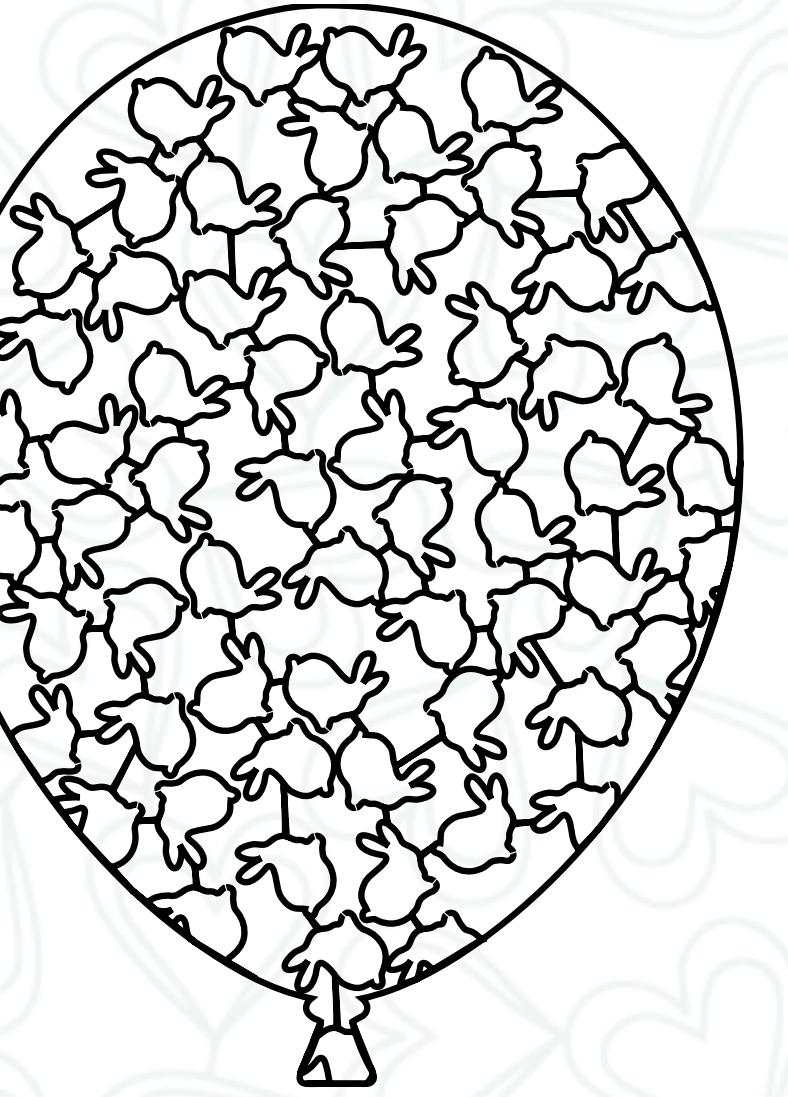
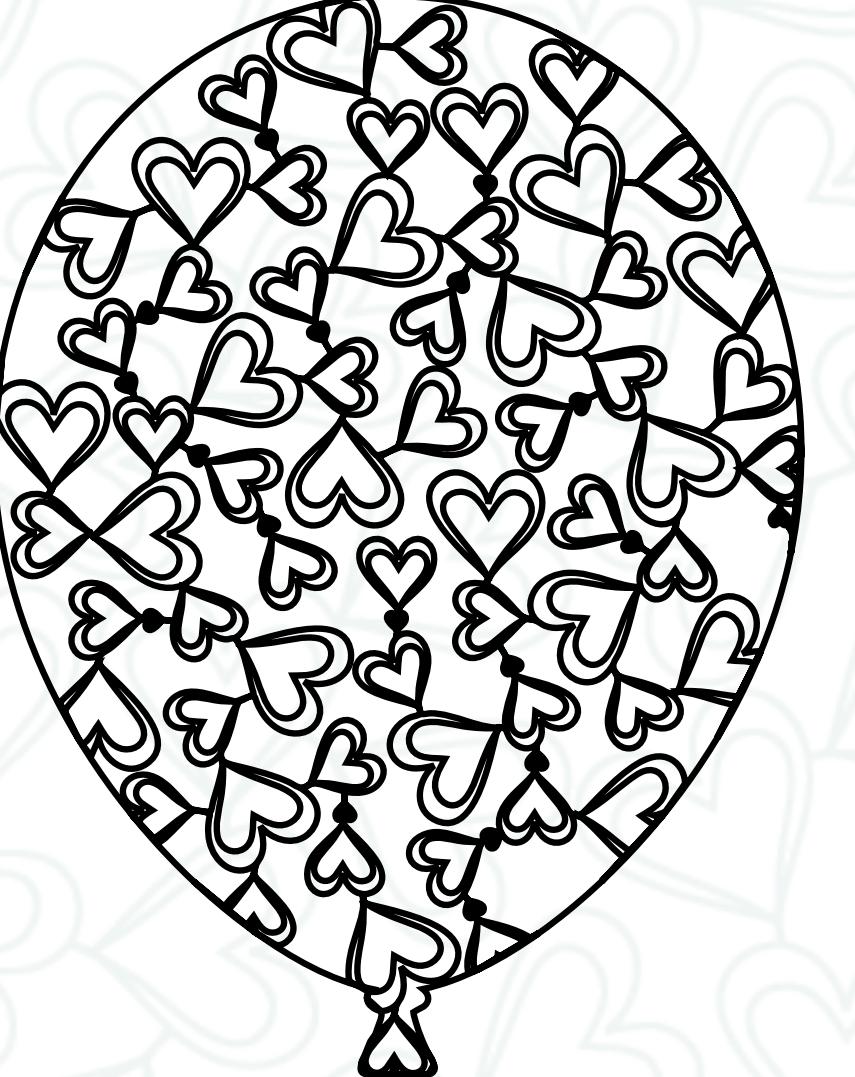


# Shape Packing

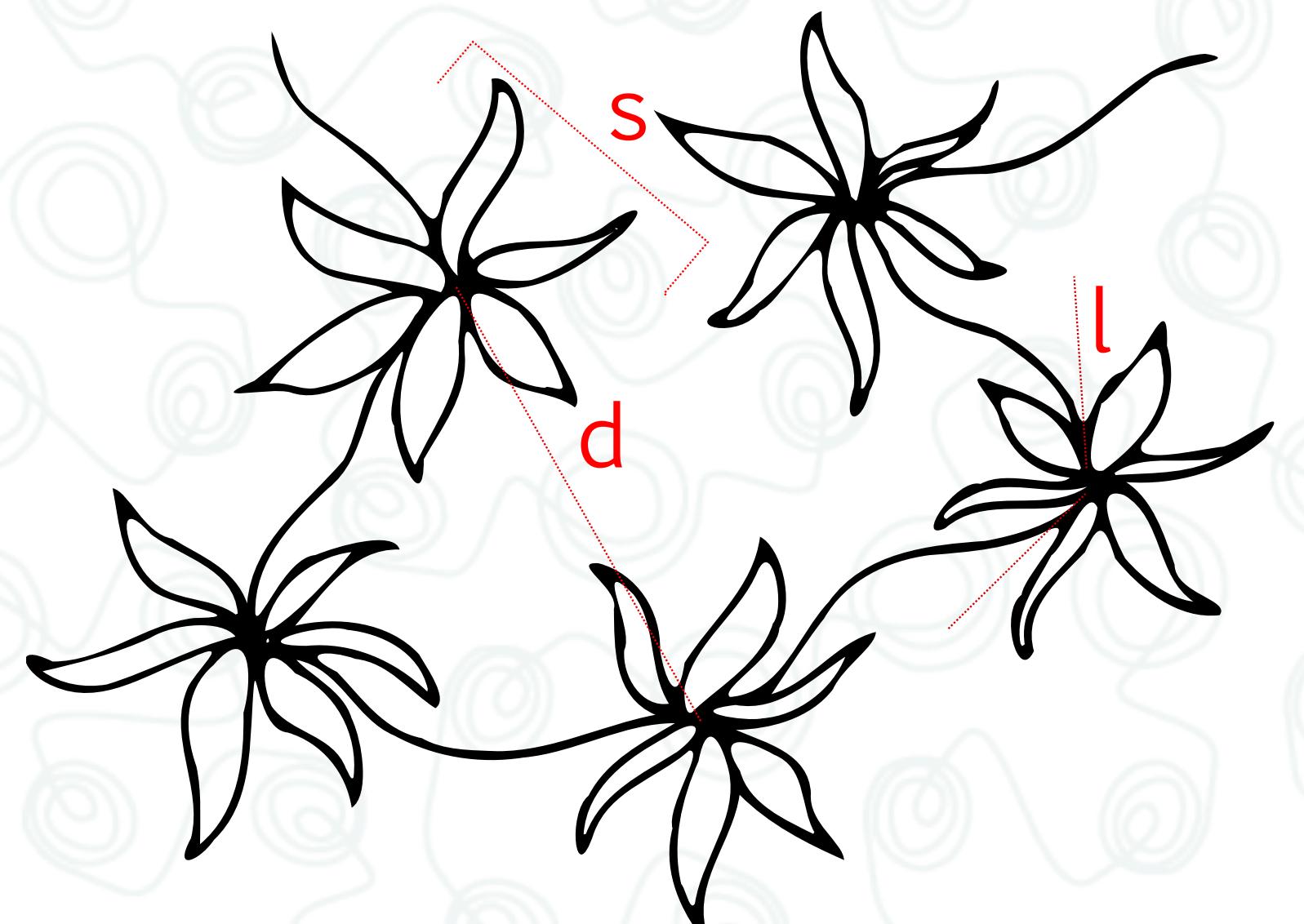
$$r_{seed} = 0.5d$$



$$r_{seed} = 0.4d$$



# Wanderer



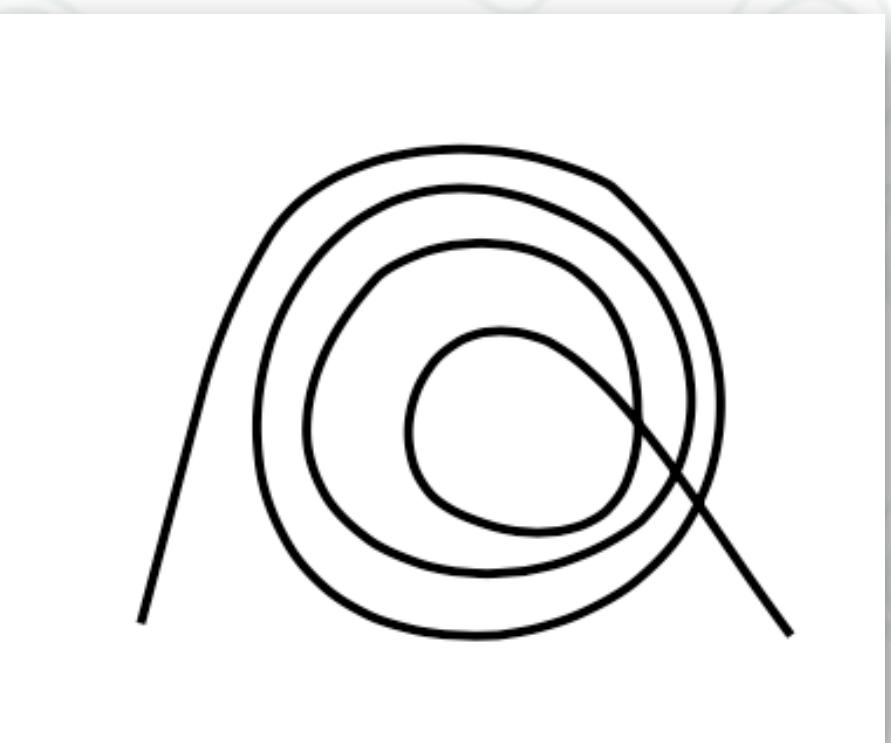
primitive is repeated and joined by another instance through wavy lines

skeleton path wanders through an open space

**Point Distribution: Any  
Skeleton Path: TSP  
Parameterization:**

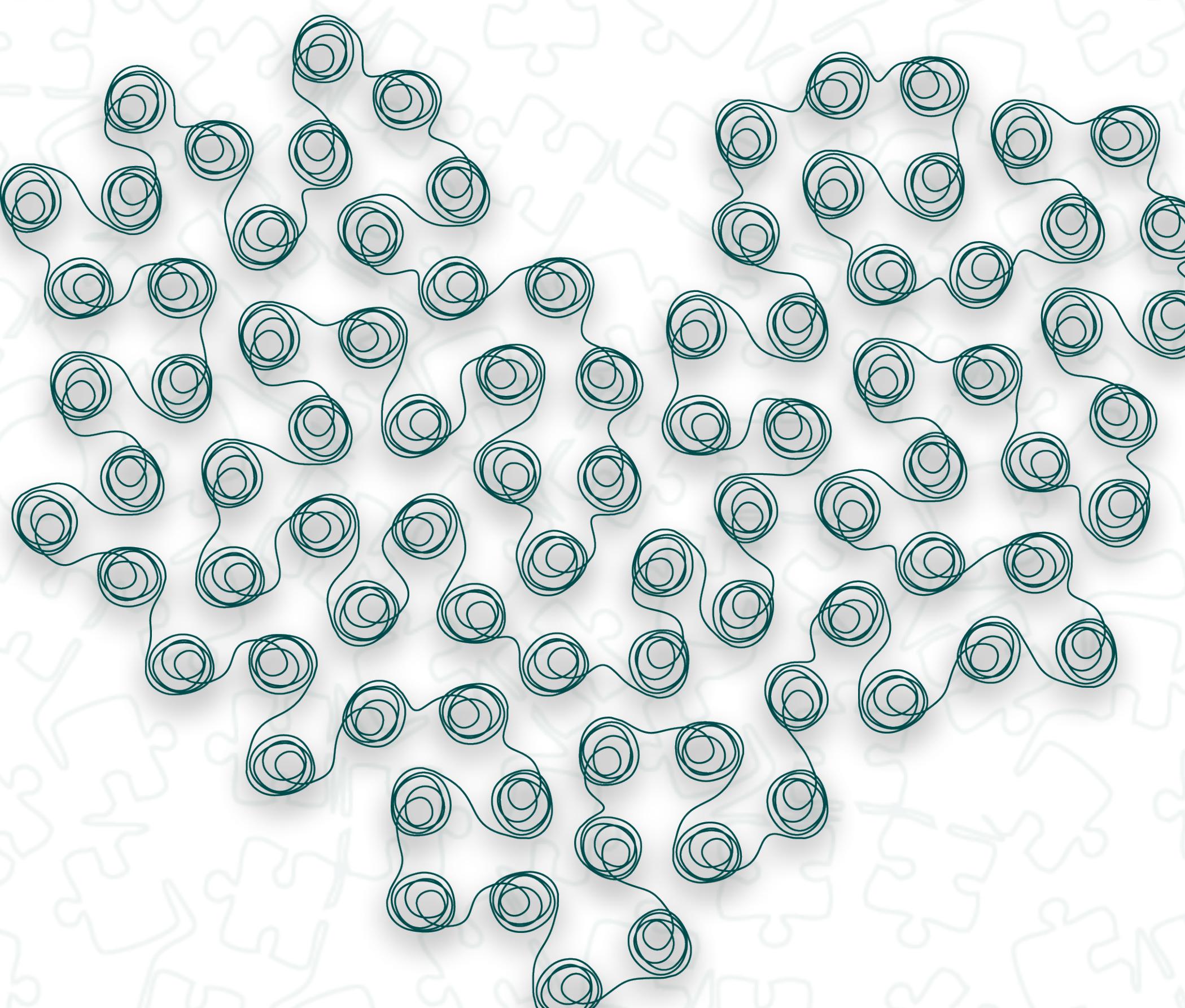
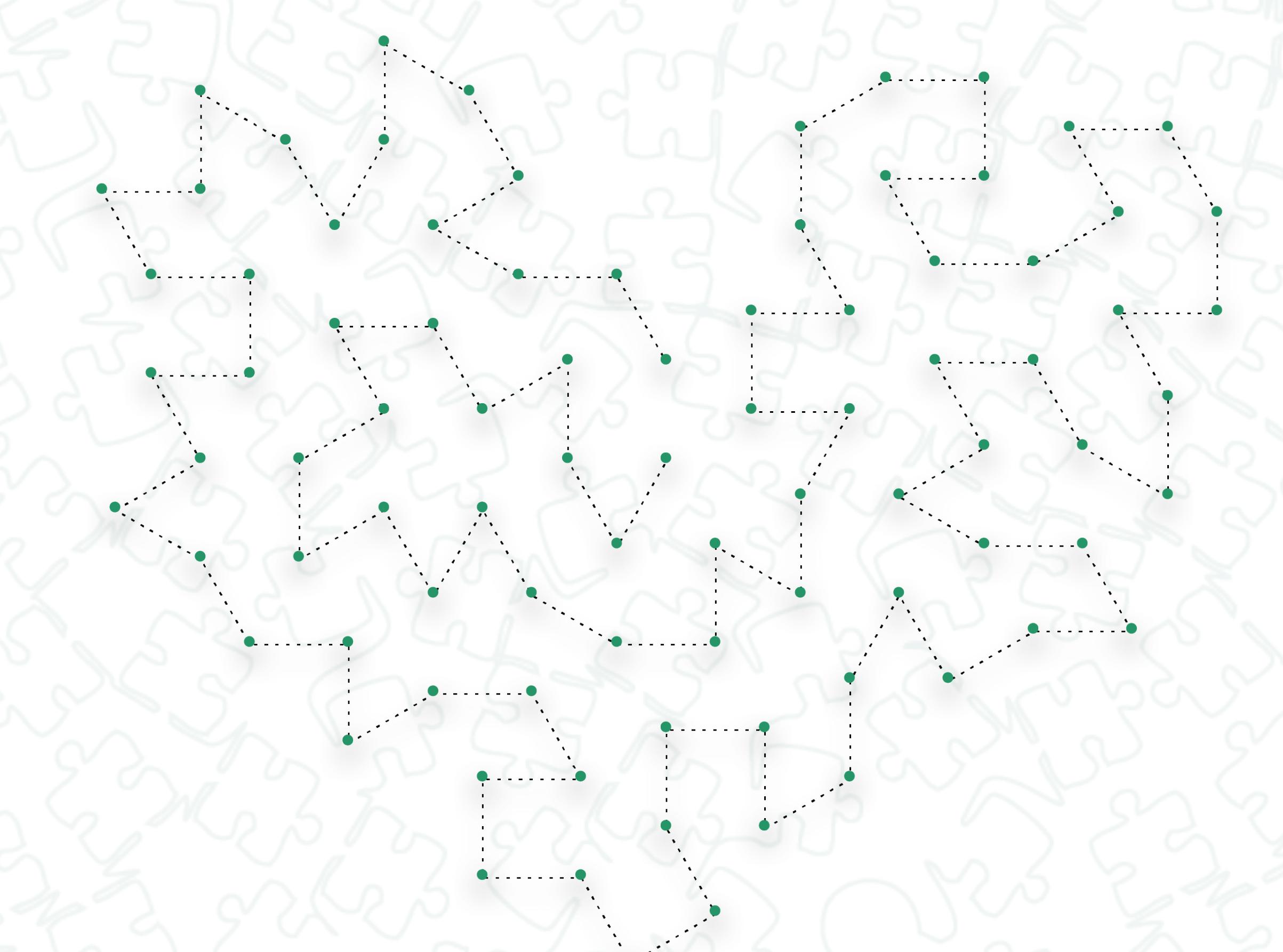
1. Point distribution distance
2. Decorative element scale
3. Tangent length

$d$   
 $s$   
 $l$

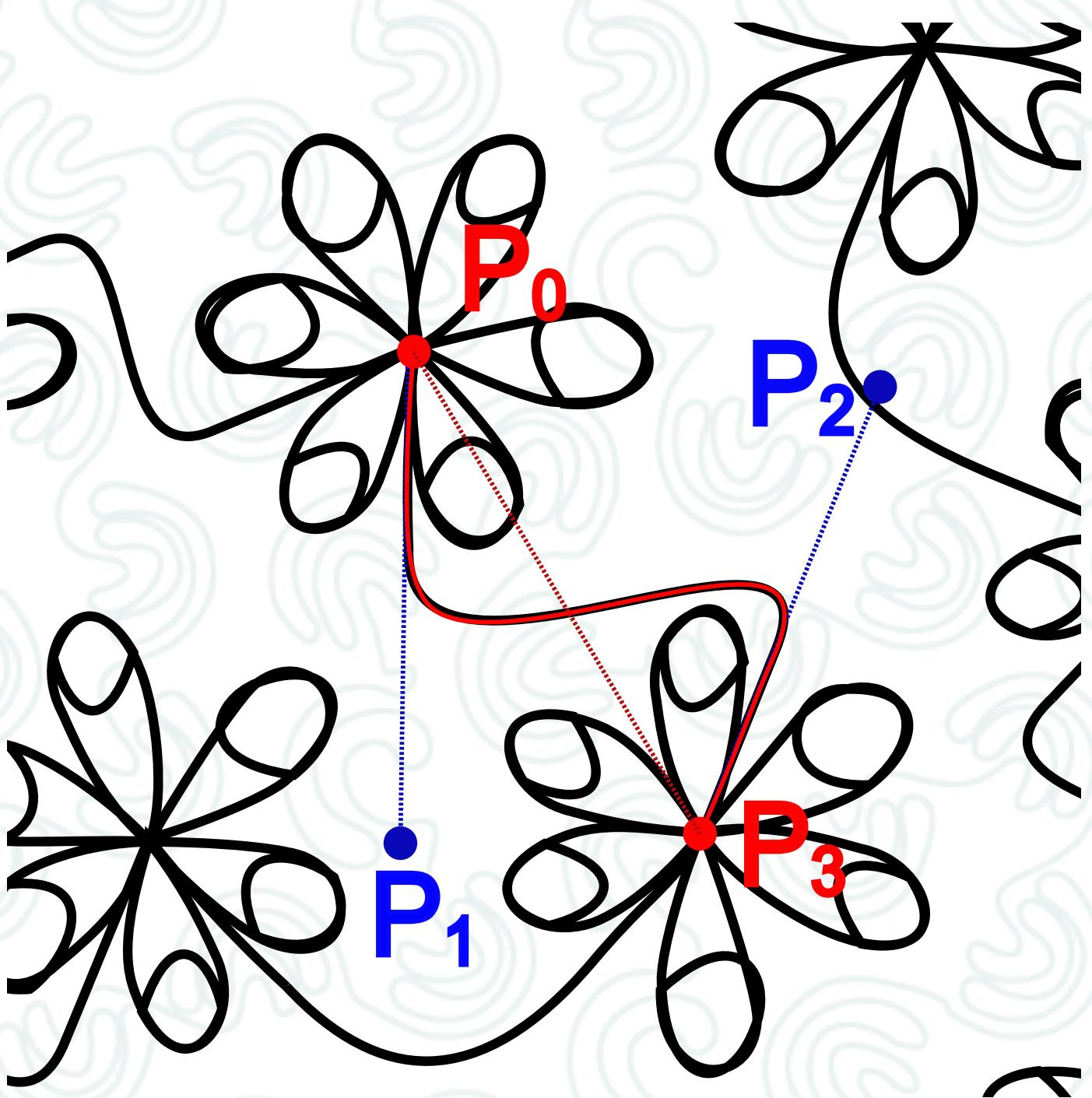
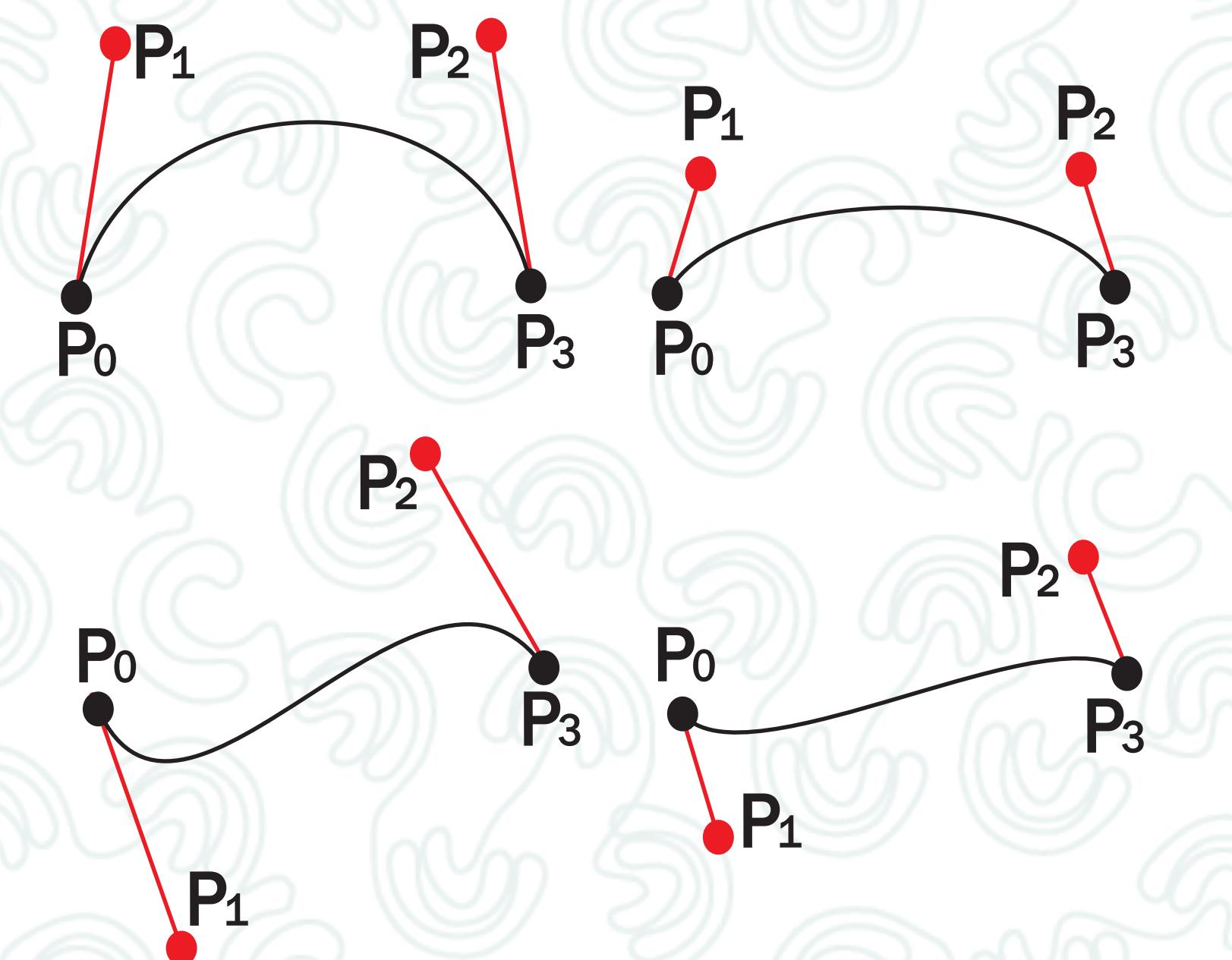


Example input design

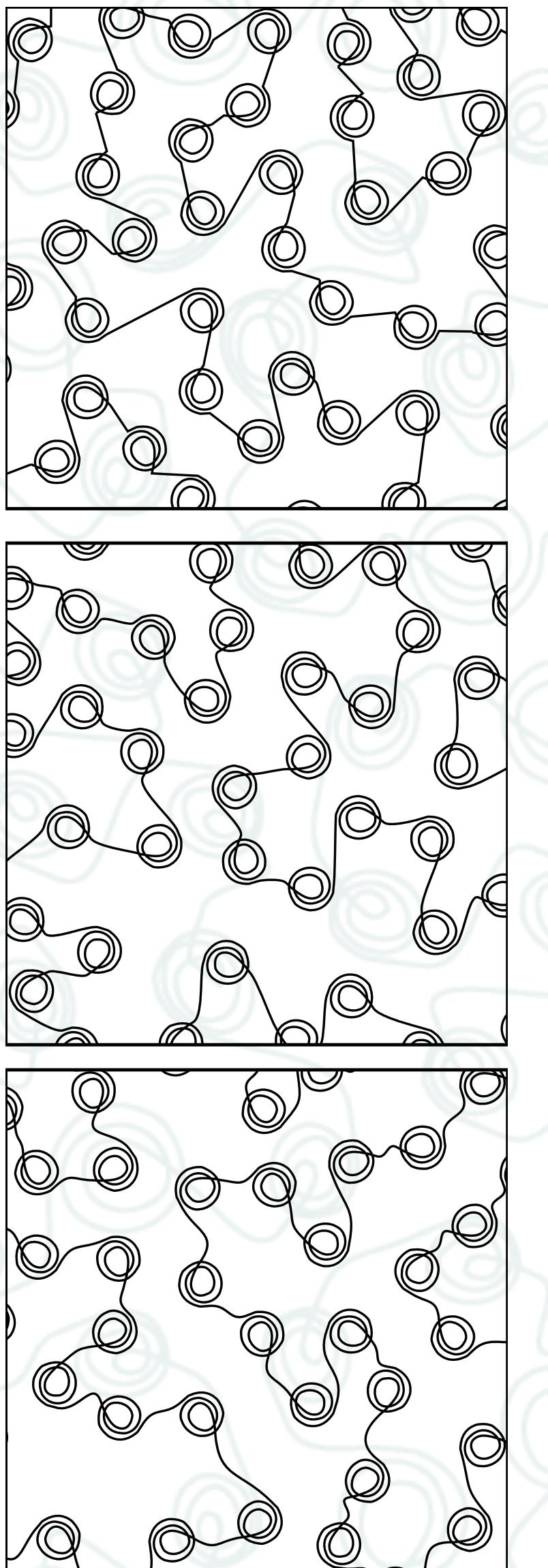
# Wanderer



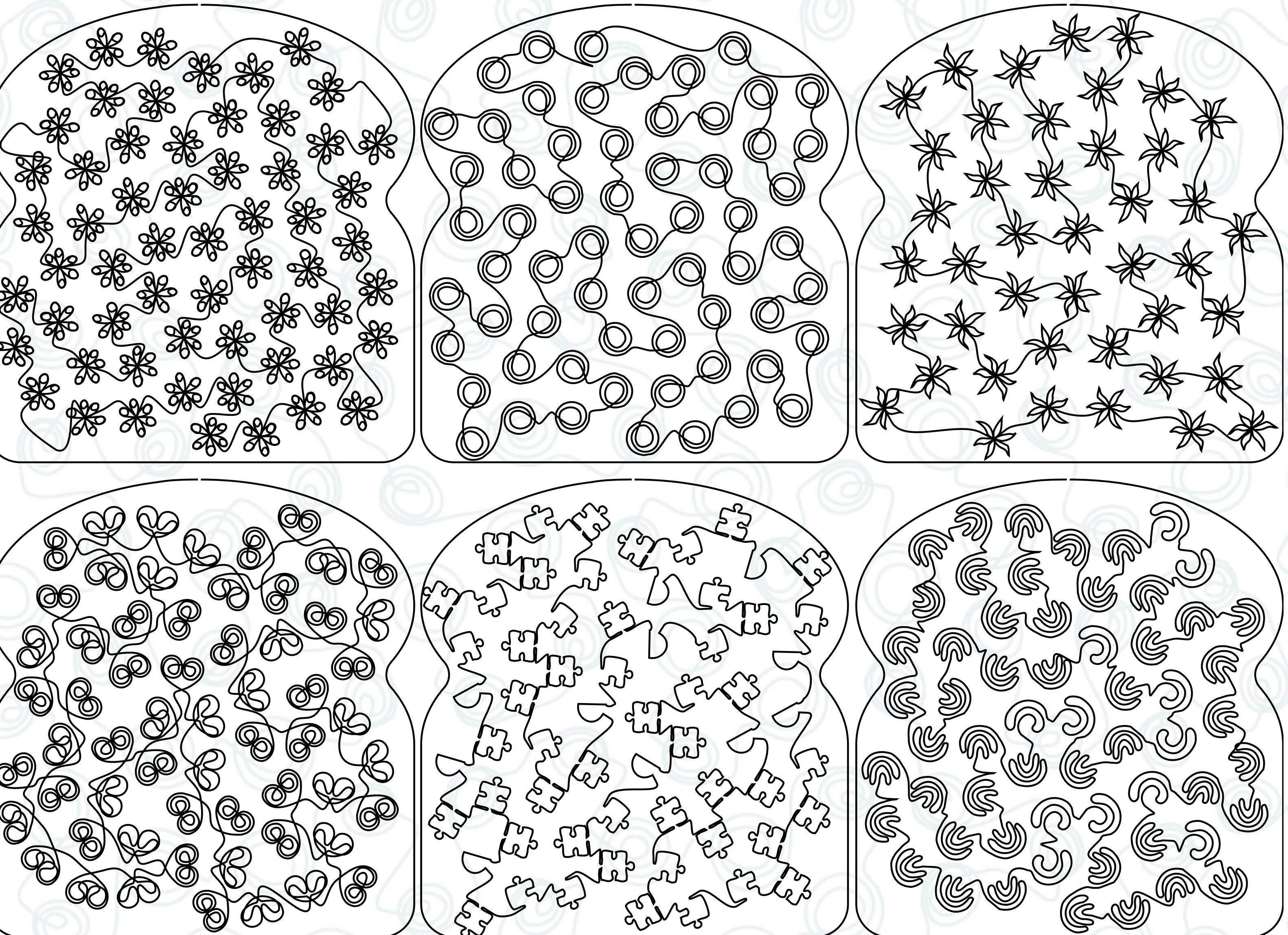
# Wanderer

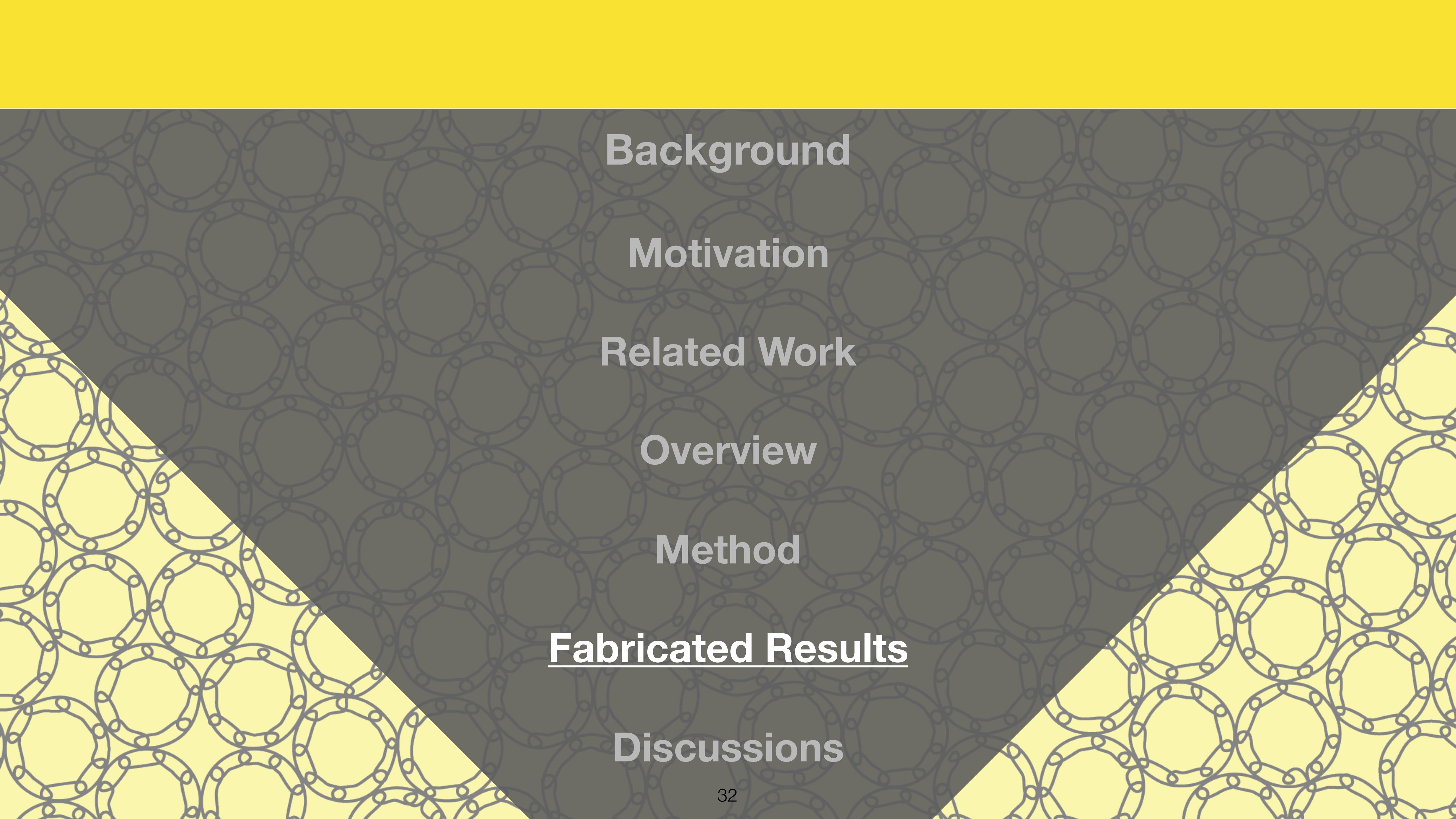


# Wanderer



increasing tangent length





**Background**

**Motivation**

**Related Work**

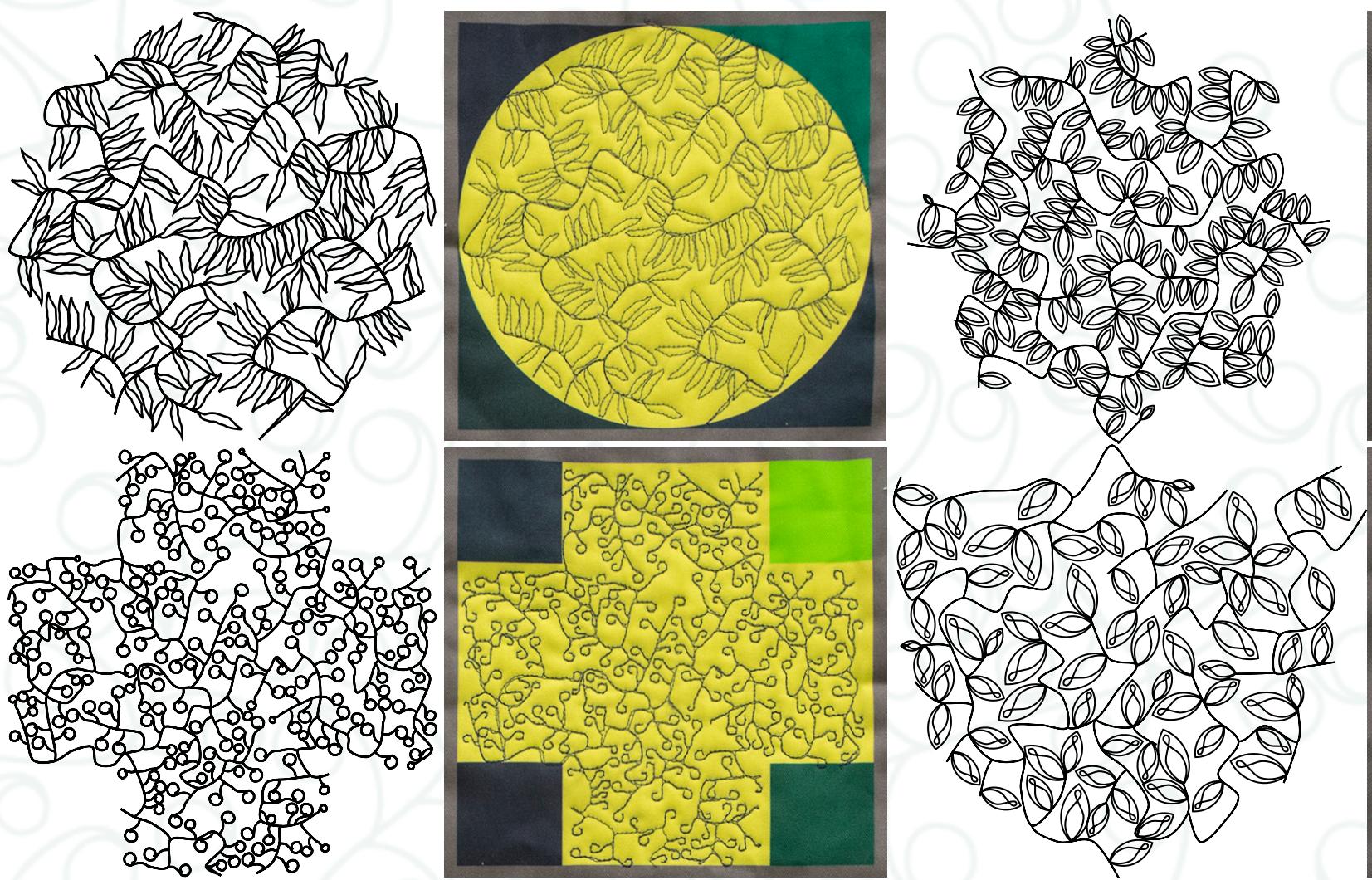
**Overview**

**Method**

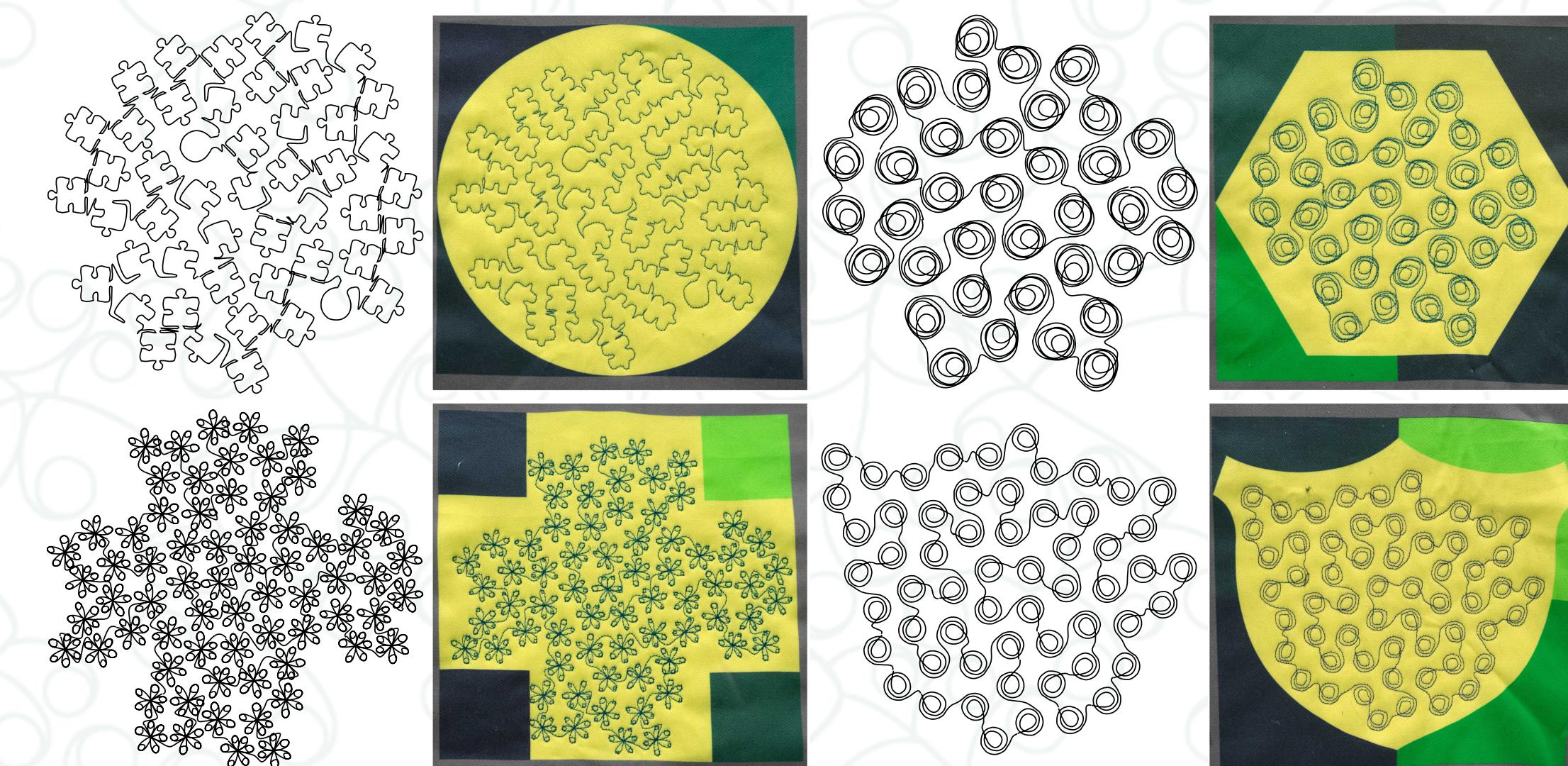
**Fabricated Results**

**Discussions**

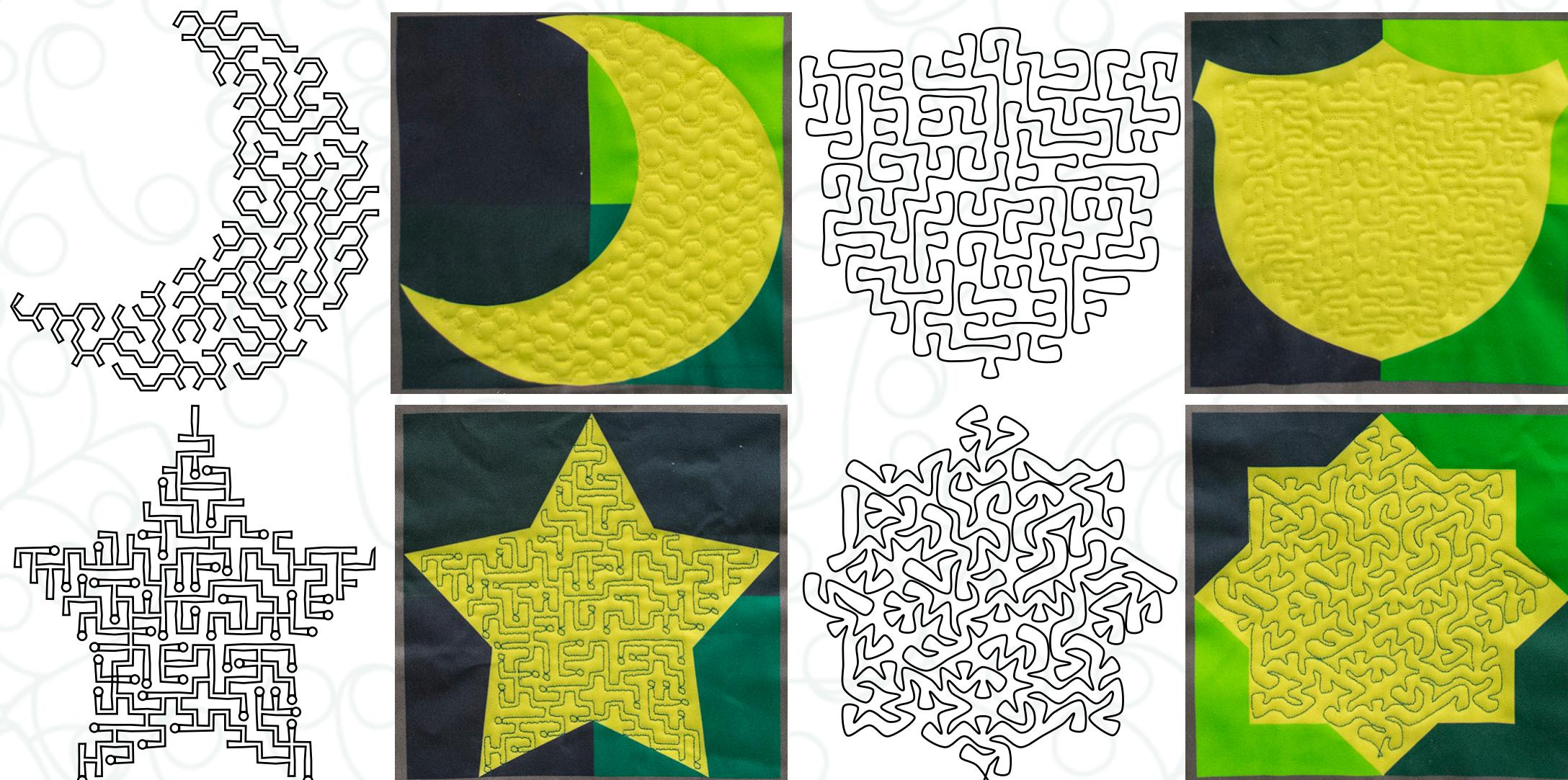
# Block Quilts



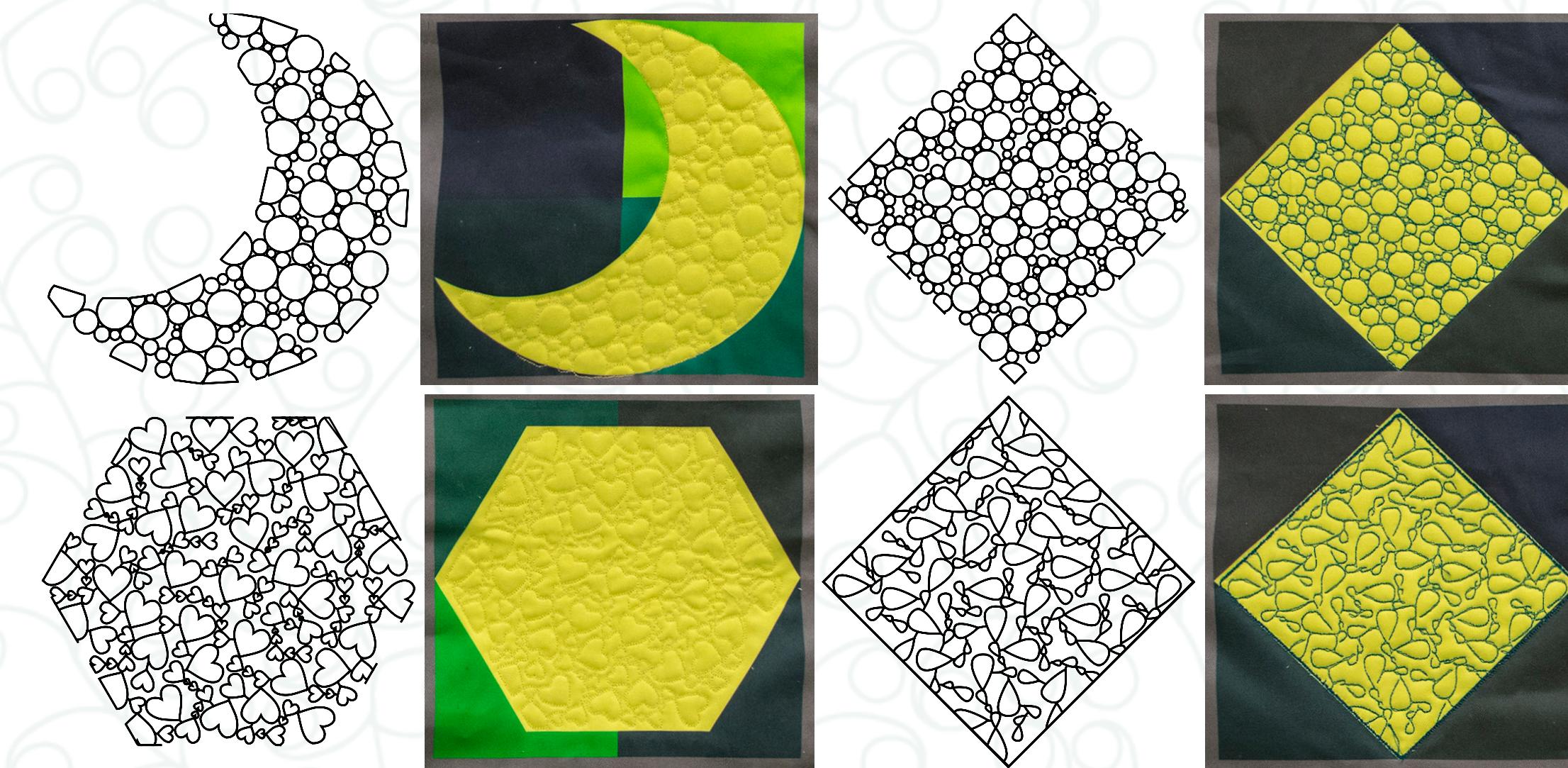
Branching



Wanderer

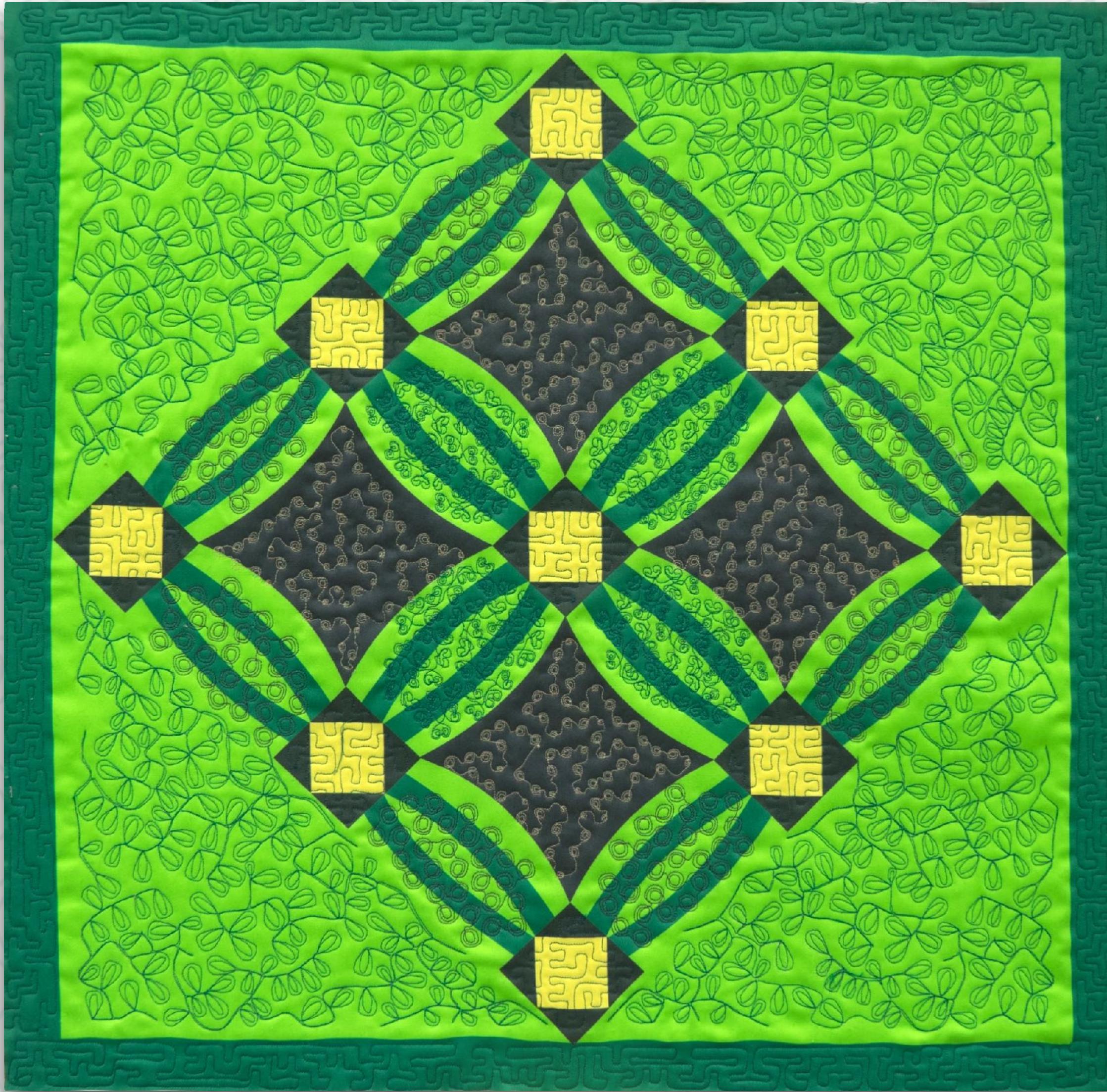


Stippling



Shape Packing

# Full Quilts

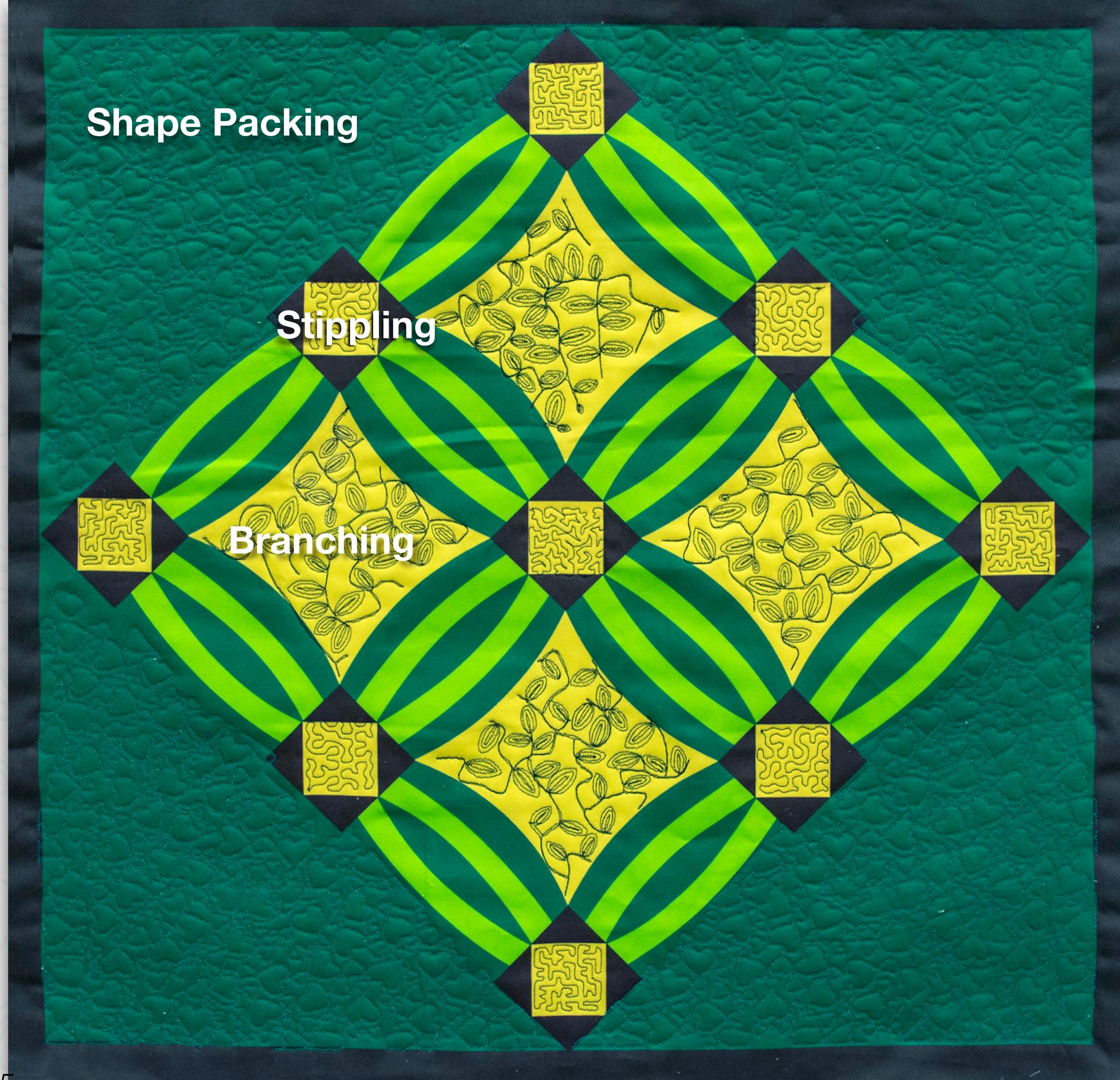
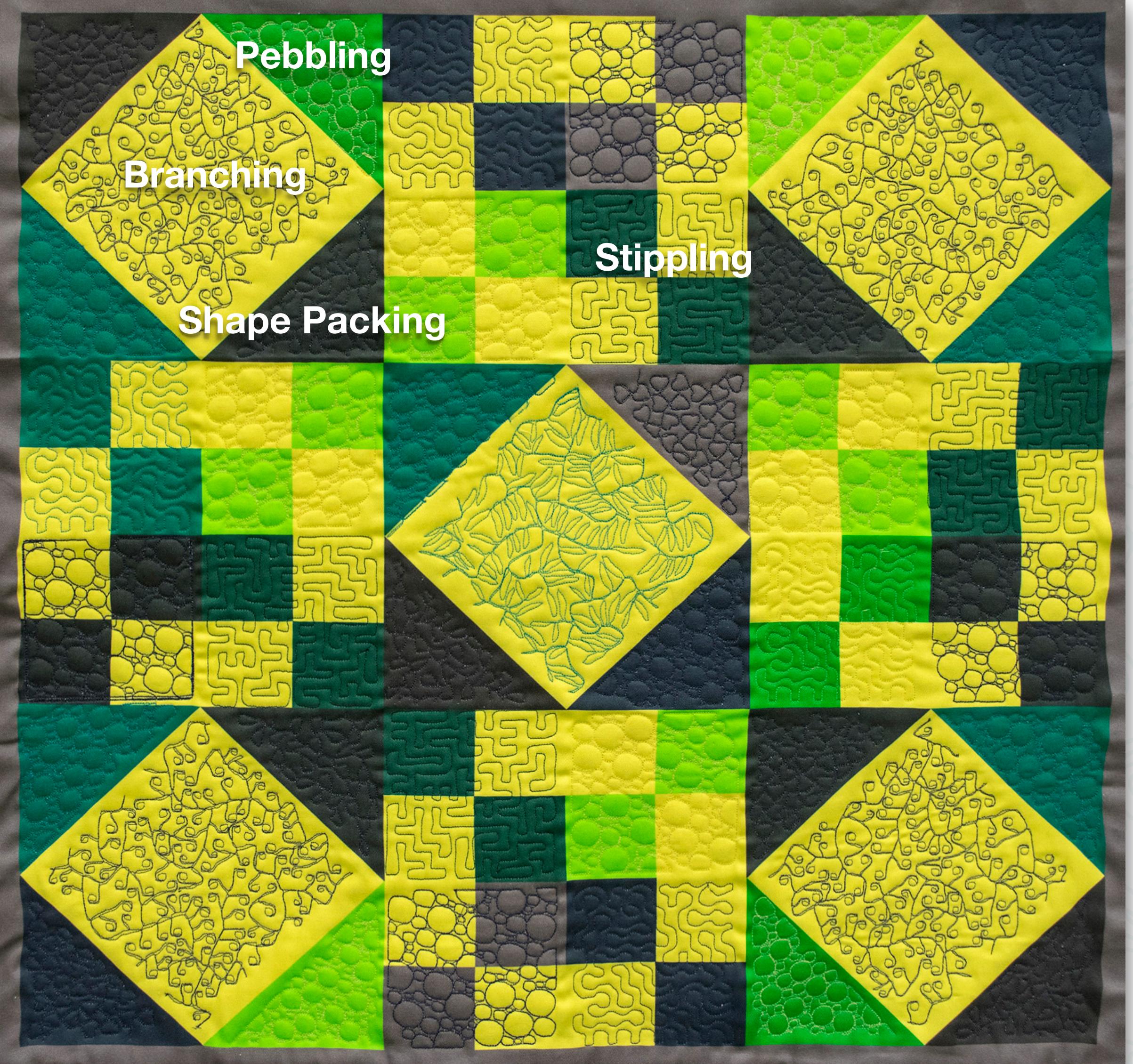


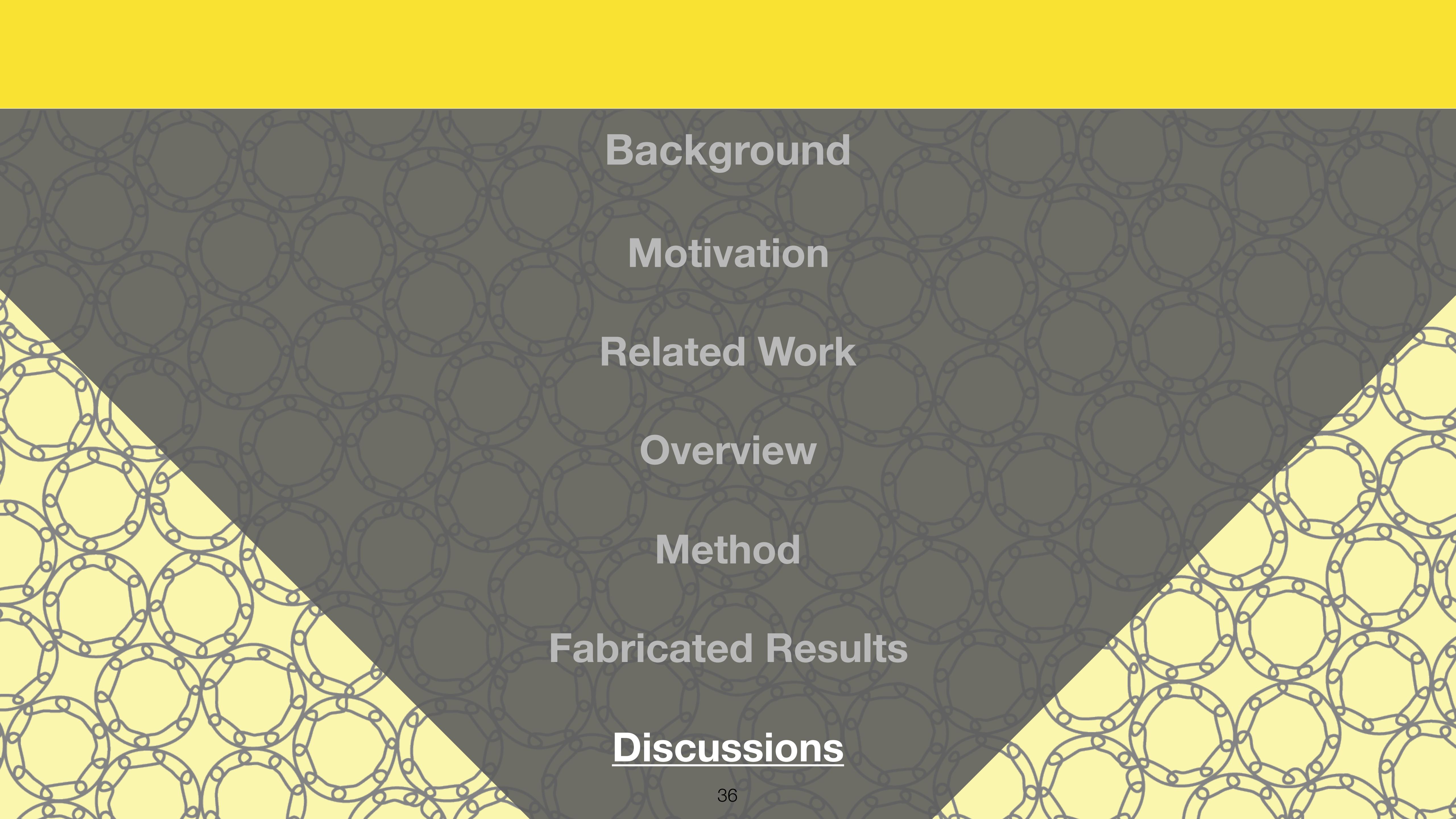
Quilt Front



Quilt Back

# Full Quilts





**Background**

**Motivation**

**Related Work**

**Overview**

**Method**

**Fabricated Results**

**Discussions**

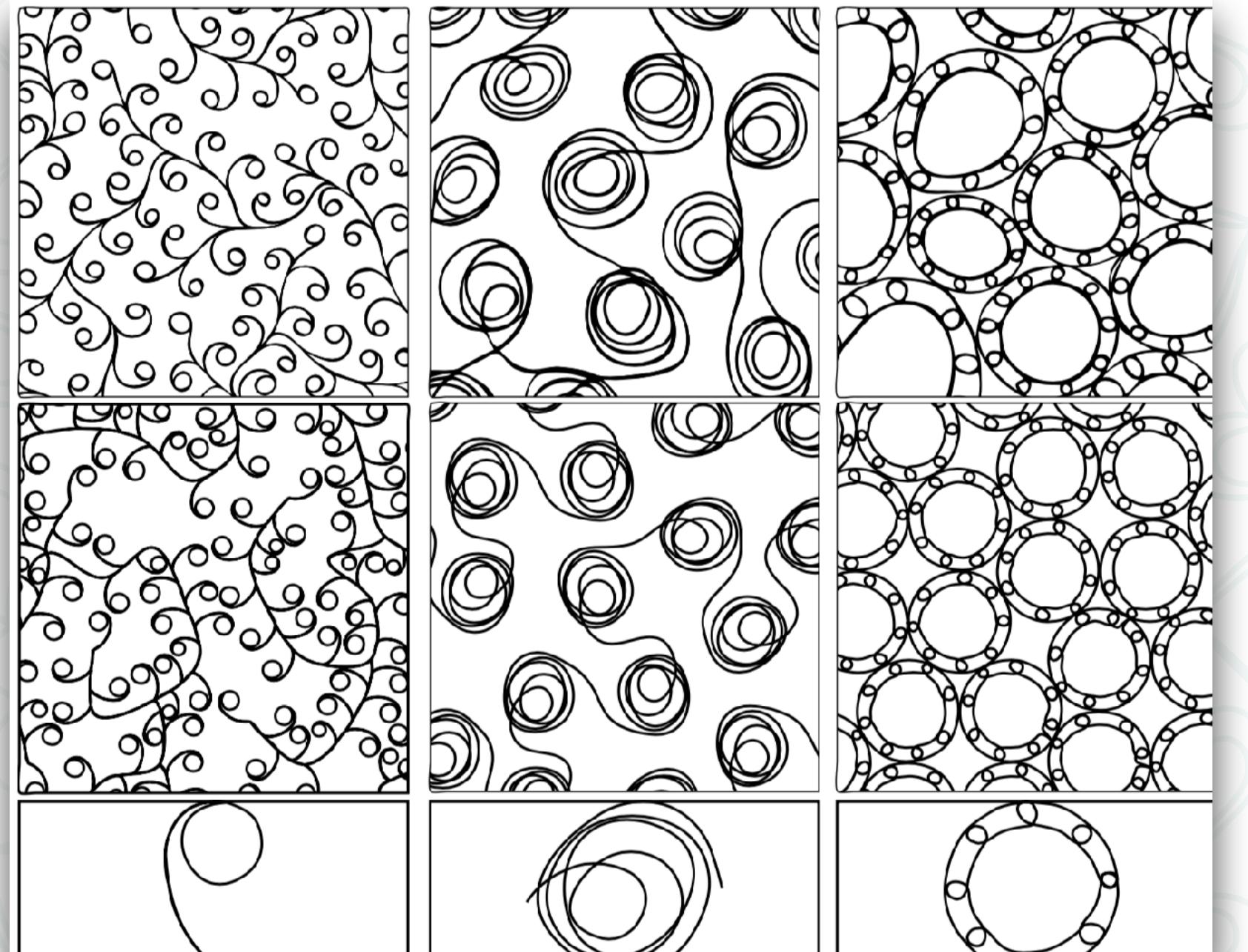
# Discussions

Contact: Yifei Li  
yifeil1@andrew.cmu.edu

**Our Java implementation is efficient**

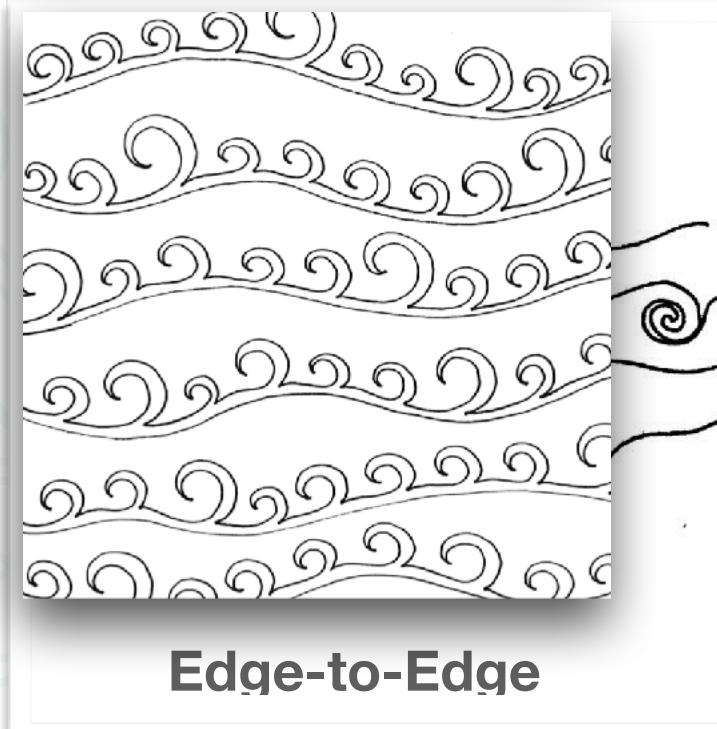
Instance	Point Set Size	Mean Generation Time
Heart Packing	118	0.62s
Leaf Branching	118	0.77s
"Dizzy" Wanderer	102	0.67s
Heart Packing	892	8.09s
Leaf Branching	892	6.59s
"Dizzy" Wanderer	755	6.63s

**Actual quilting patterns have more natural variability**



**Our algorithm: Branching, Wanderers, Shape Packing**

**Future work: Four categories from Cameli's categorization**



Edge-to-Edge



Emerging



Climber



Back-and-forth Echoina

**An interactive interface with real-time preview**