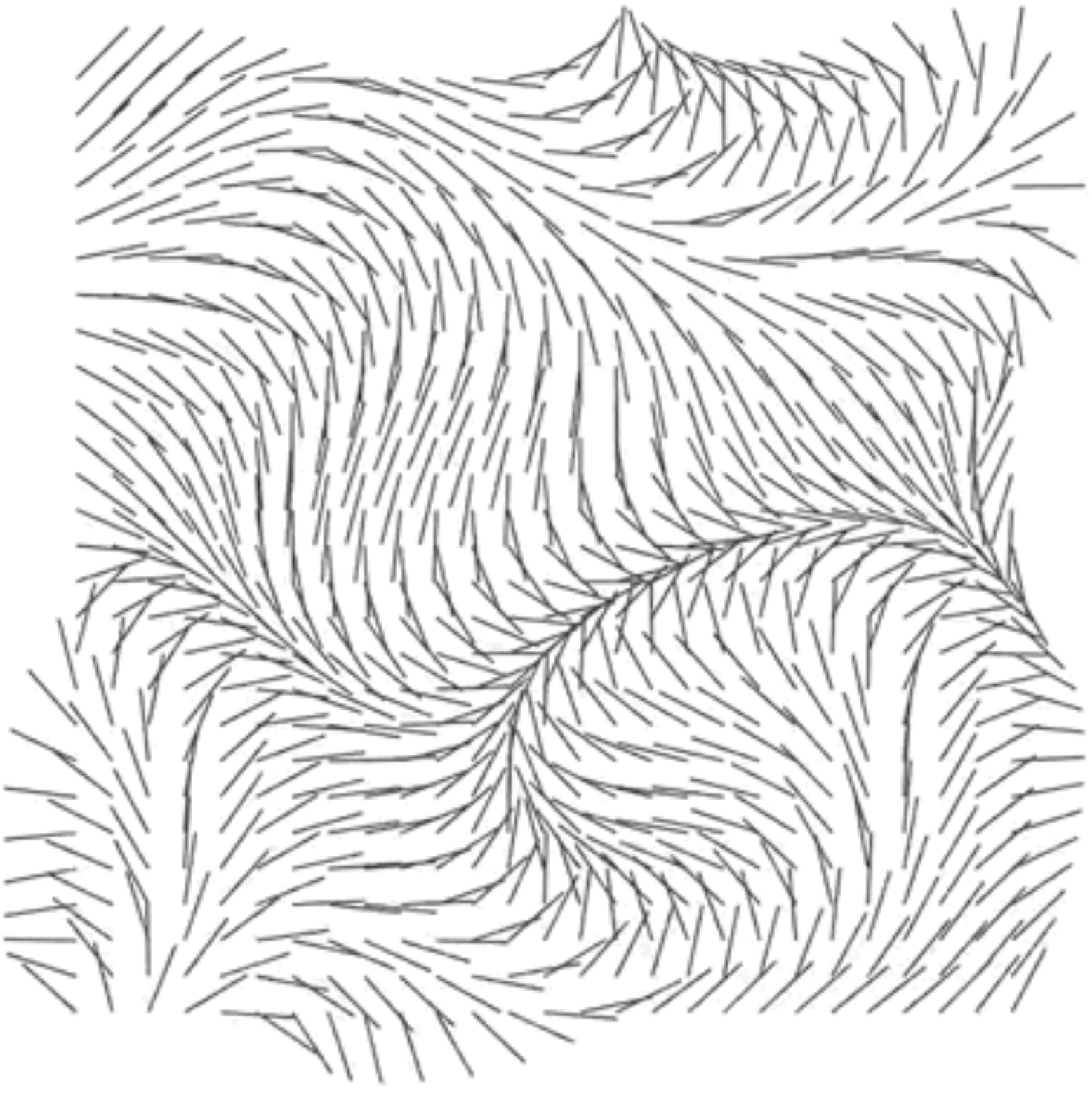
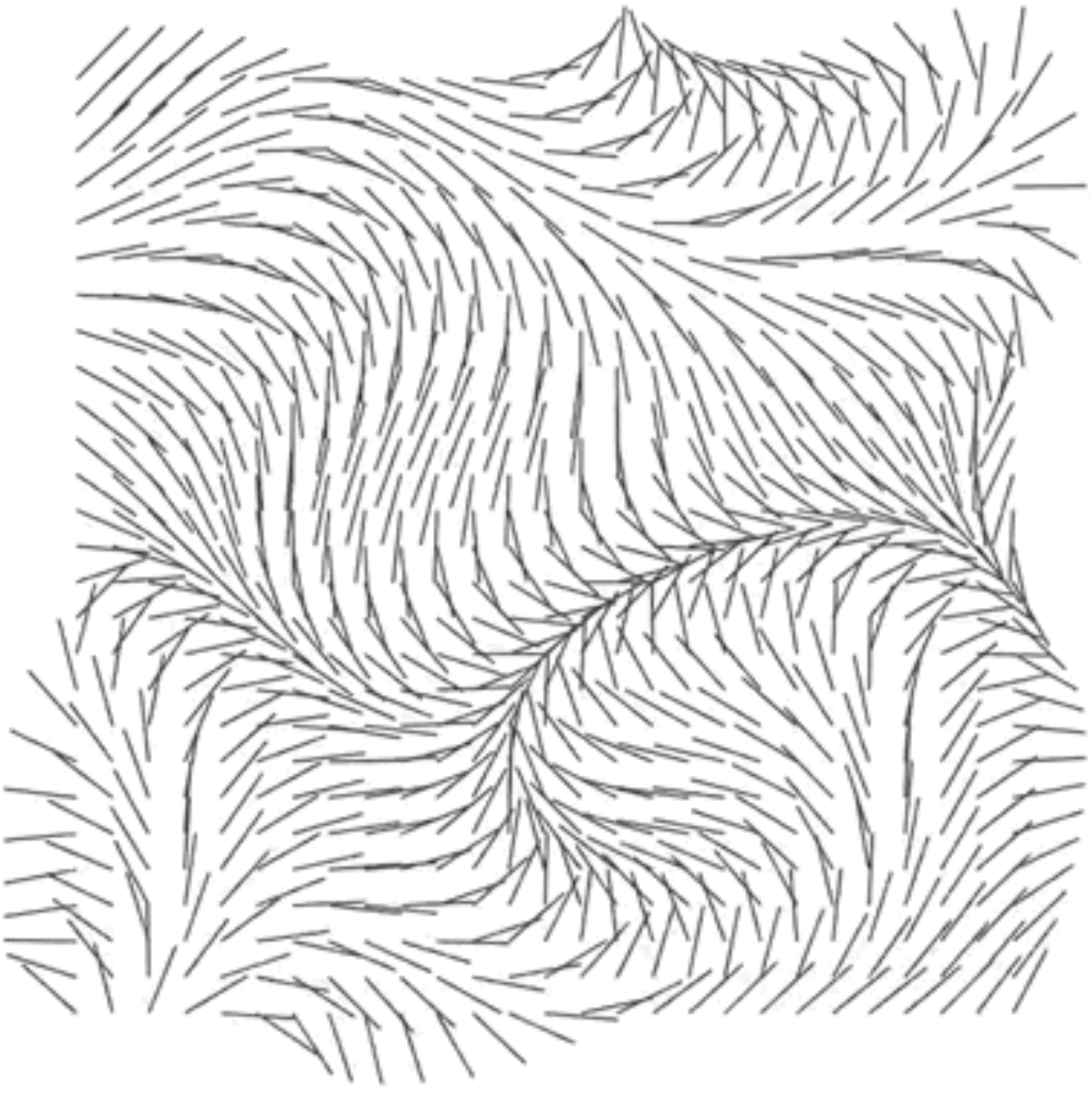
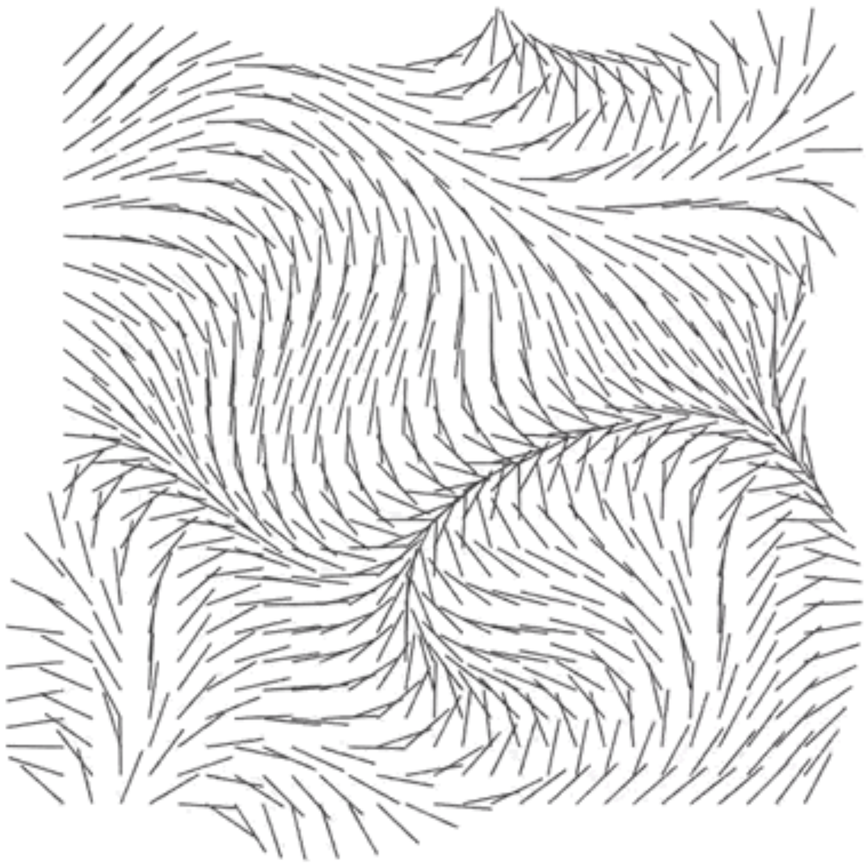


```
val radians = glm.simplex(
  Vec4(
    x = u,
    y = v,
    z = 5f * cos(TWO_PI * time / 20f),
    w = 5f * sin(TWO_PI * time / 20f)
  )
) * TWO_PI

val endX = startX + (r * sin(radians))
val endY = startY + (r * cos(radians))
drawLine(
  start = Offset(startX, startY),
  end = Offset(endX, endY),
  ...
)
```

Animated Flow Fields!

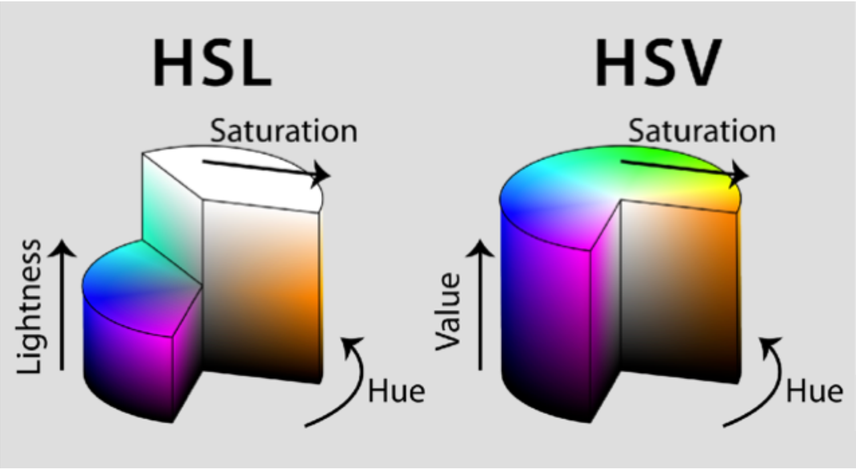


Animated Flow Fields!



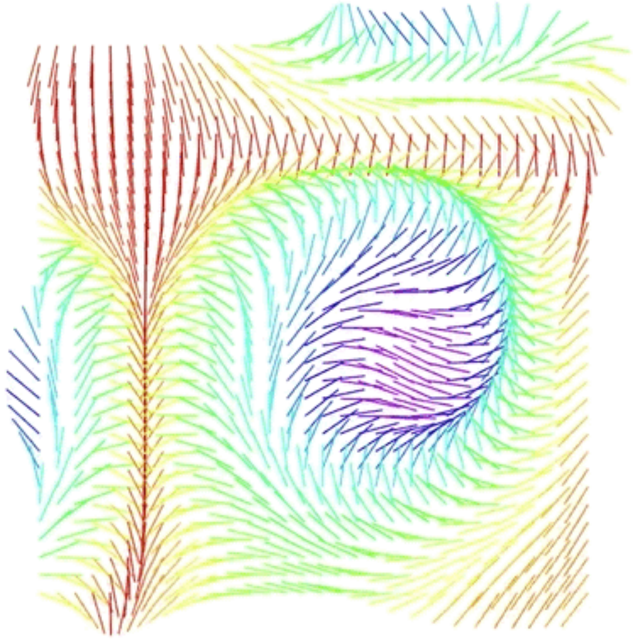
```
val radians = glm.simplex(
  Vec4(
    x = u,
    y = v,
    z = 5f * cos(TWO_PI * time / 20f),
    w = 5f * sin(TWO_PI * time / 20f)
  )
) * TWO_PI

val endX = startX + (r * sin(radians))
val endY = startY + (r * cos(radians))
drawLine(
  start = Offset(startX, startY),
  end = Offset(endX, endY),
  ...
)
```

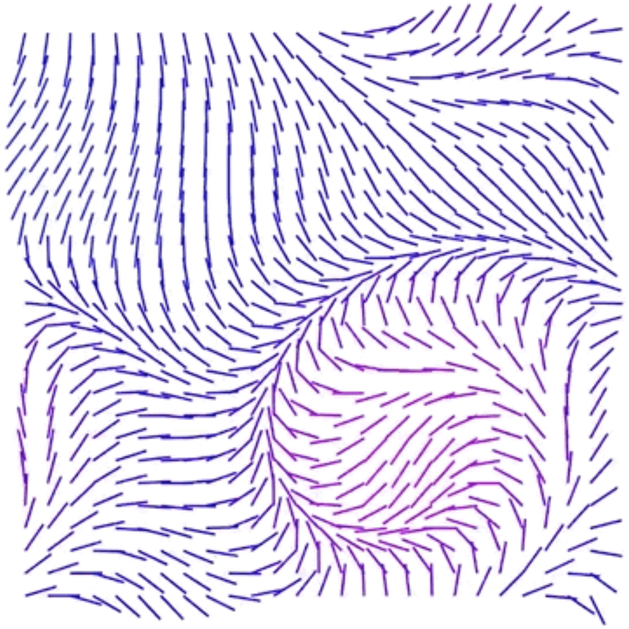
Color



wikipedia.org

- HSL, HSV use hue
- hue is in [0, 360] -> like angles?
- map noise to hue ranges 🌈





```
// Compute hue based on noise
val hue = (noise * 360f).absoluteValue
// OR
val hue = map(noise, -1f, 1f, 170f, 300f)

// Use hue in hsv
val color = Color.hsv(hue, saturation=1f, value=1f)
drawLine(color = color, ...)
```