

# Curl Quantization for Automatic Placement of Knit Singularities

Rahul Mitra<sup>(BU)</sup>, Mattéo Couplet<sup>(BU)</sup>, Tongtong Wang<sup>(LS)</sup>,  
Megan Hoffman<sup>(NEU)</sup>, Kui Wu<sup>(LS)</sup>, Edward Chien<sup>(BU)</sup>

(BU)



(LS)



(NEU)



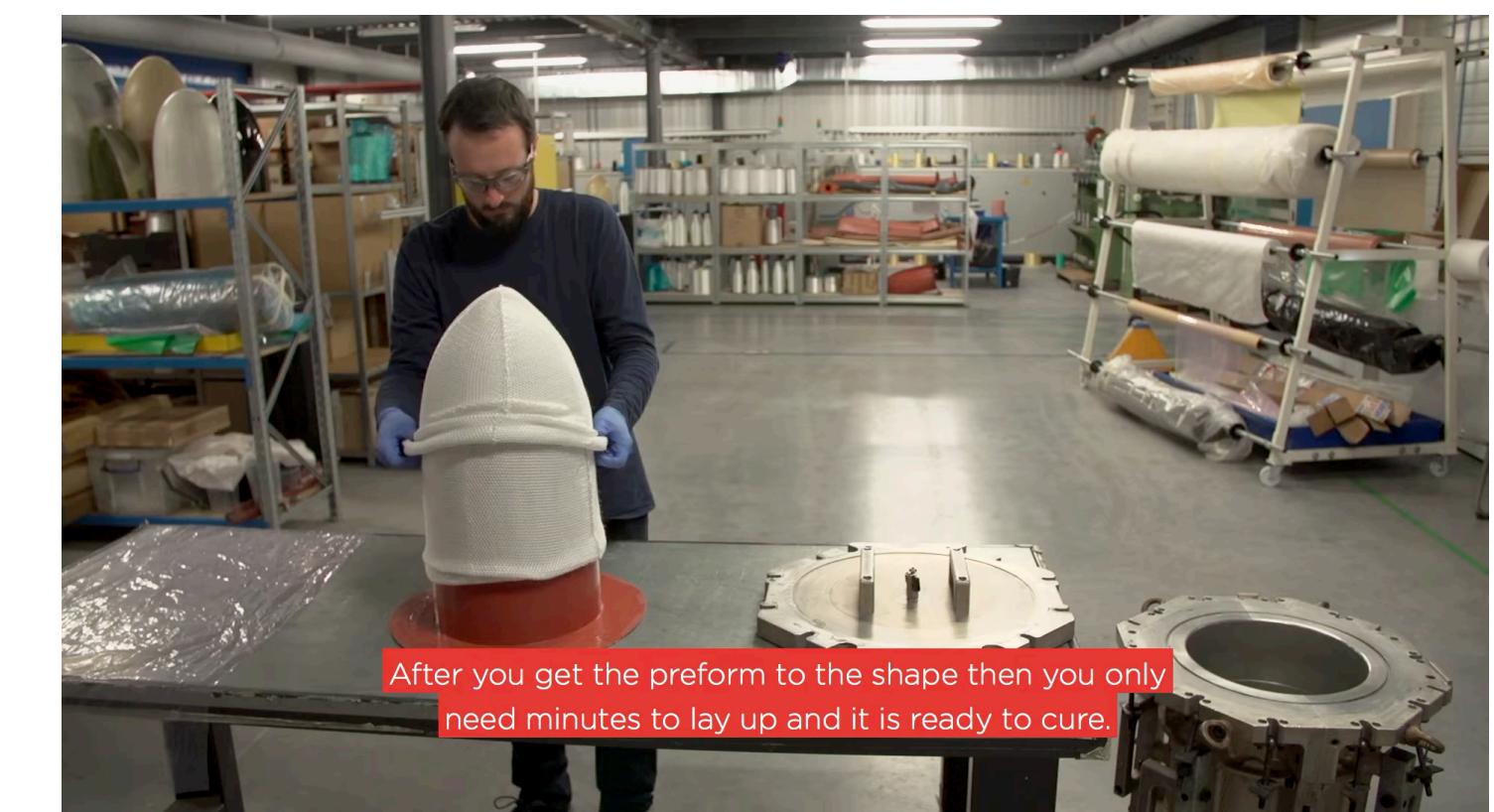


# Knitting applications

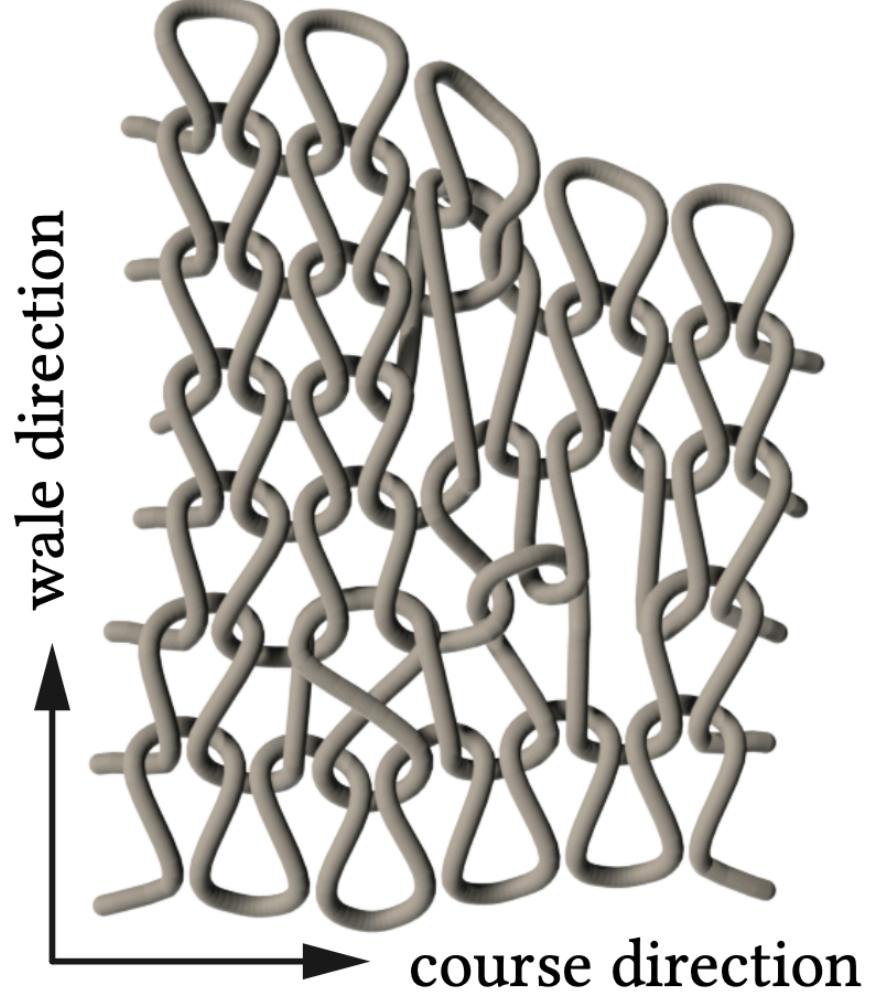
## Garments



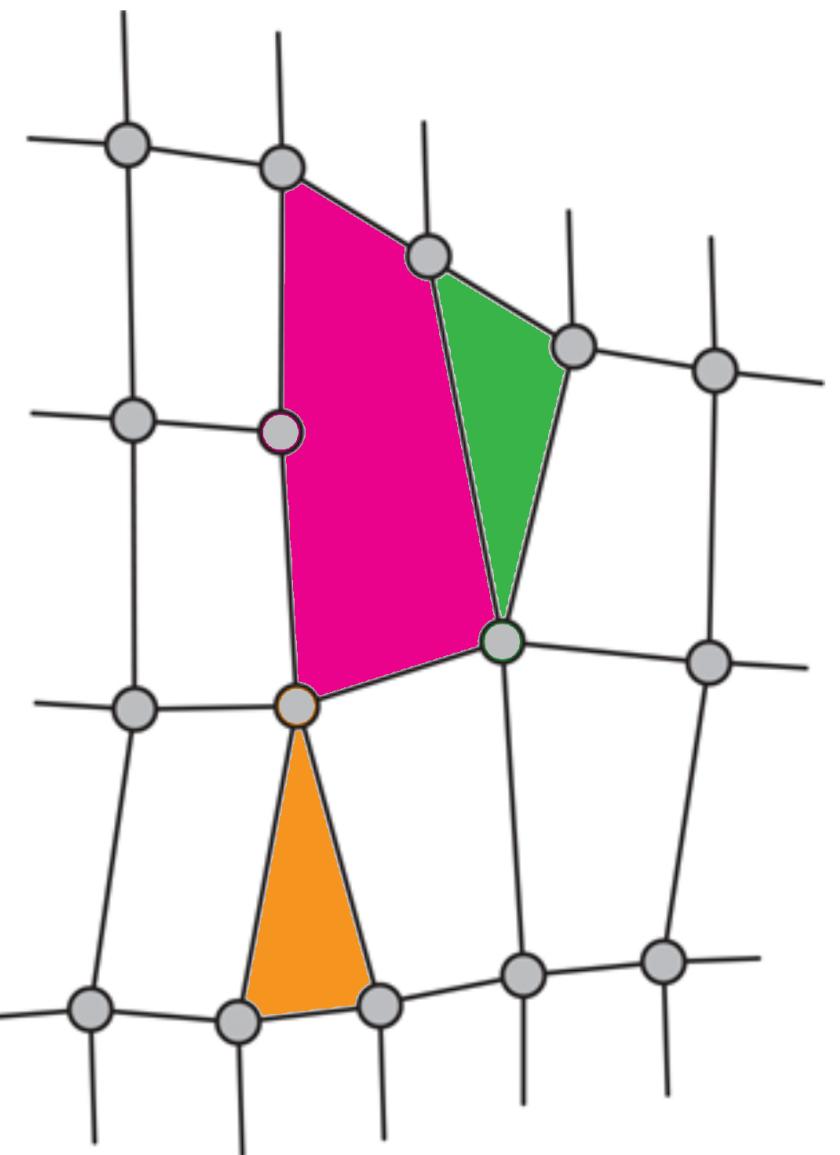
## Composites



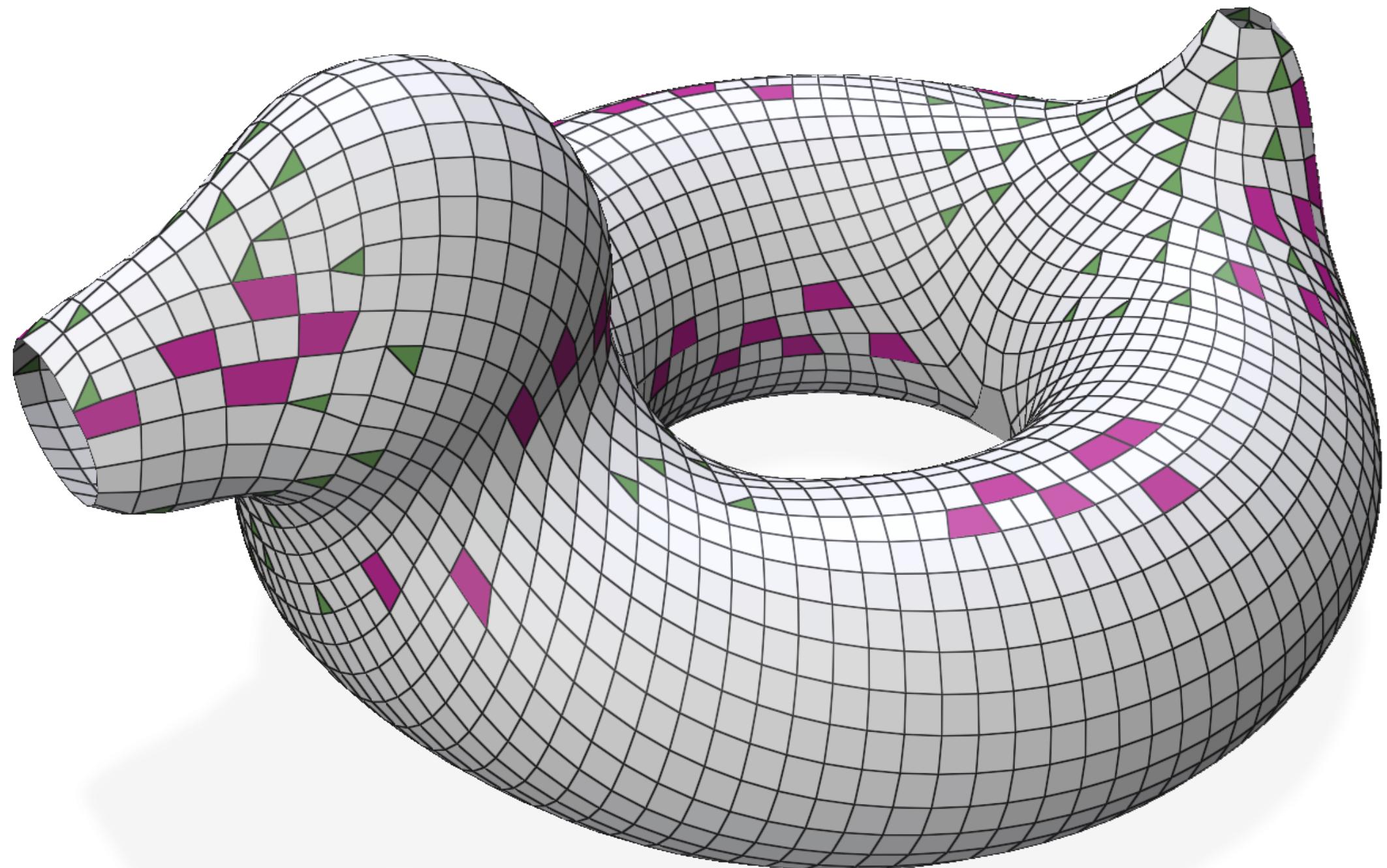
# Yarn structure is abstracted by a *knit graph*



Yarn



Knit graph



Stitch irregularities:

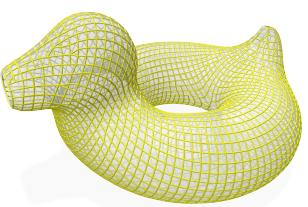
short row end

increase

decrease

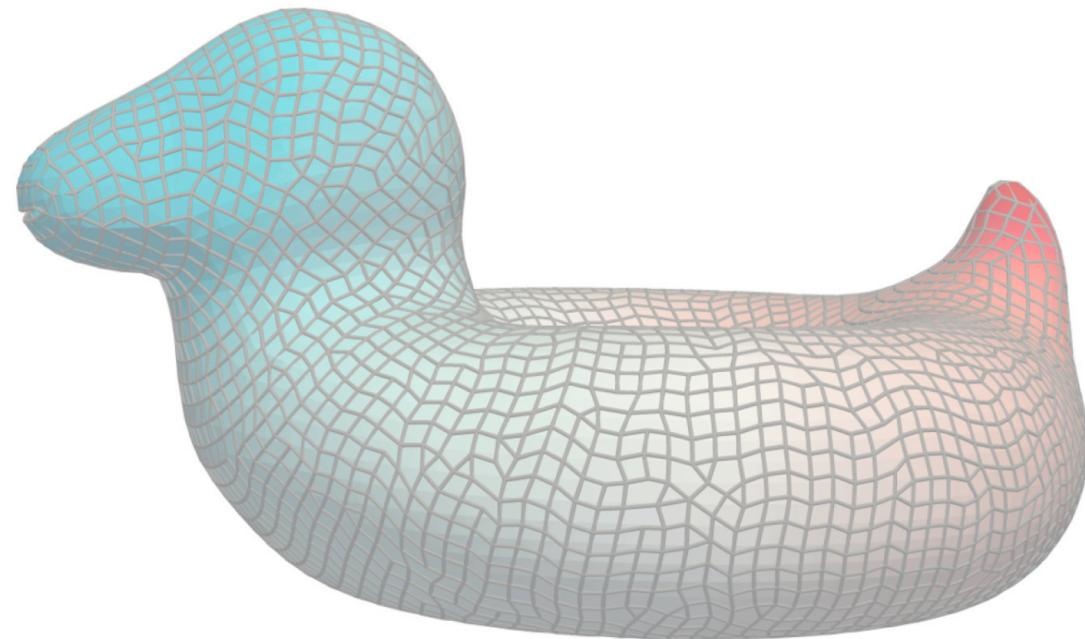
induce curvature

Goal: given target shape,  
place stitch irregularities  
to make knit graph  
as uniform as possible



# What we bring

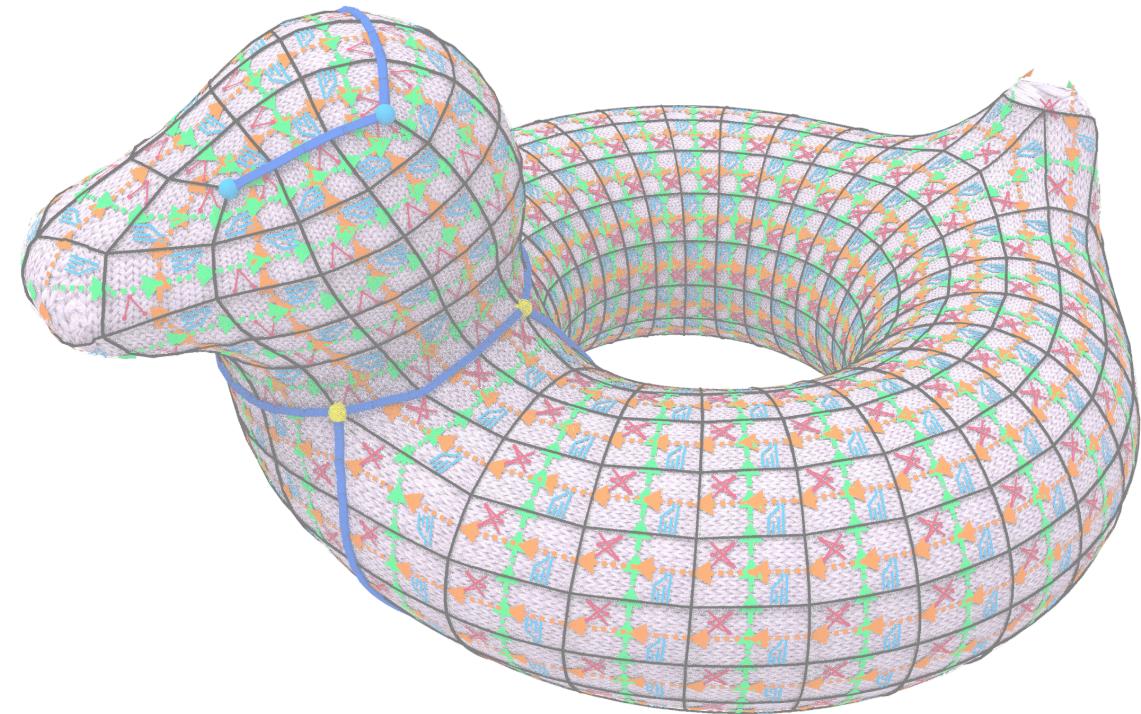
## State of the art



*AutoKnit [Narayanan18]:*  
Not global, lack of user control



*KnitKit [Nader21]:*  
No manufacturability  
guarantee



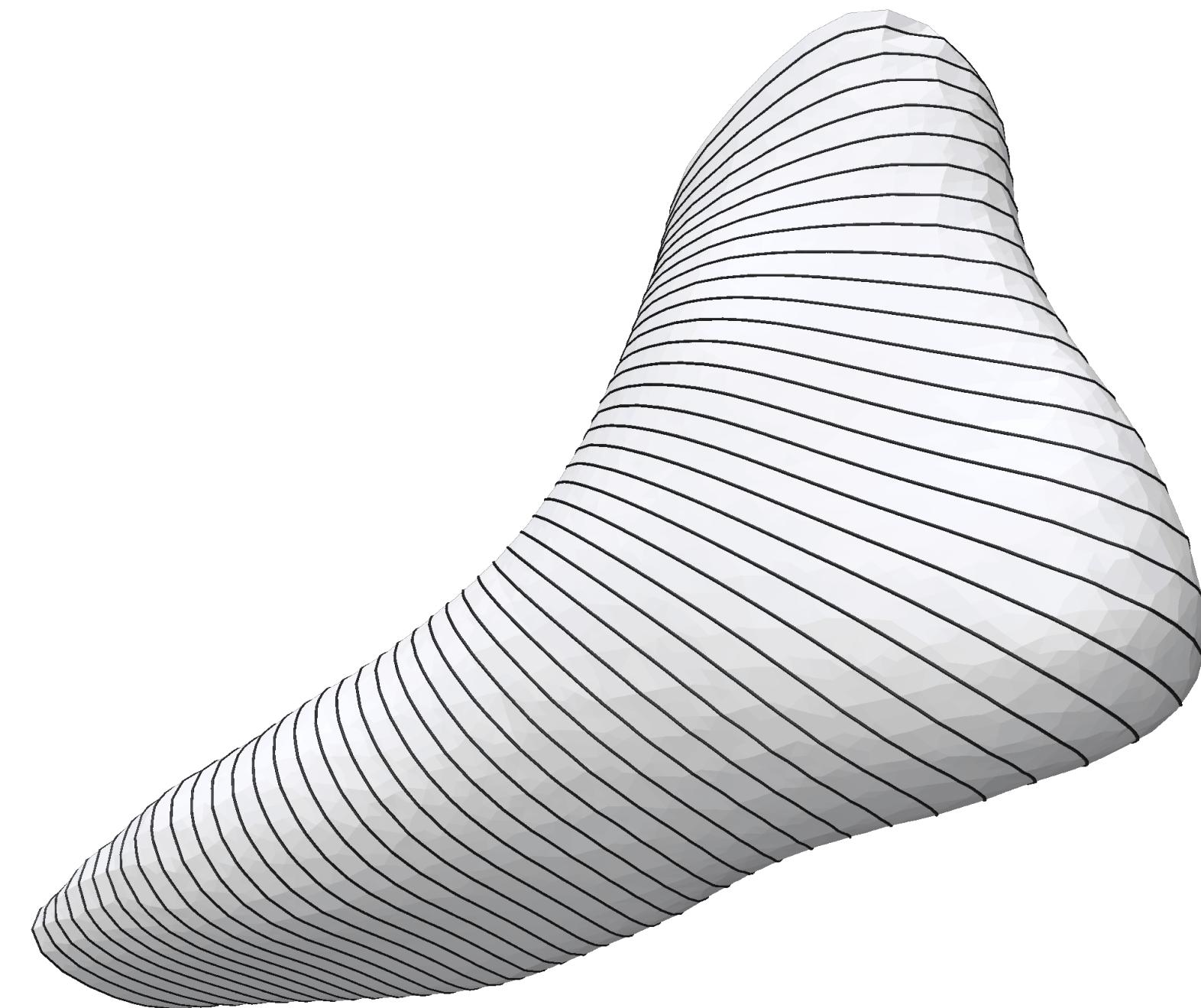
*Knit Templates [Jones22]:*  
No stitch-level  
singularity placement

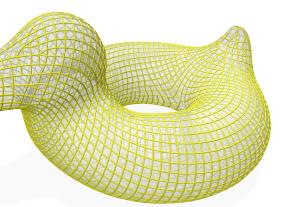


*Helix-free Stripes [Mitra23]:*  
Costly mixed-integer solve

## This paper

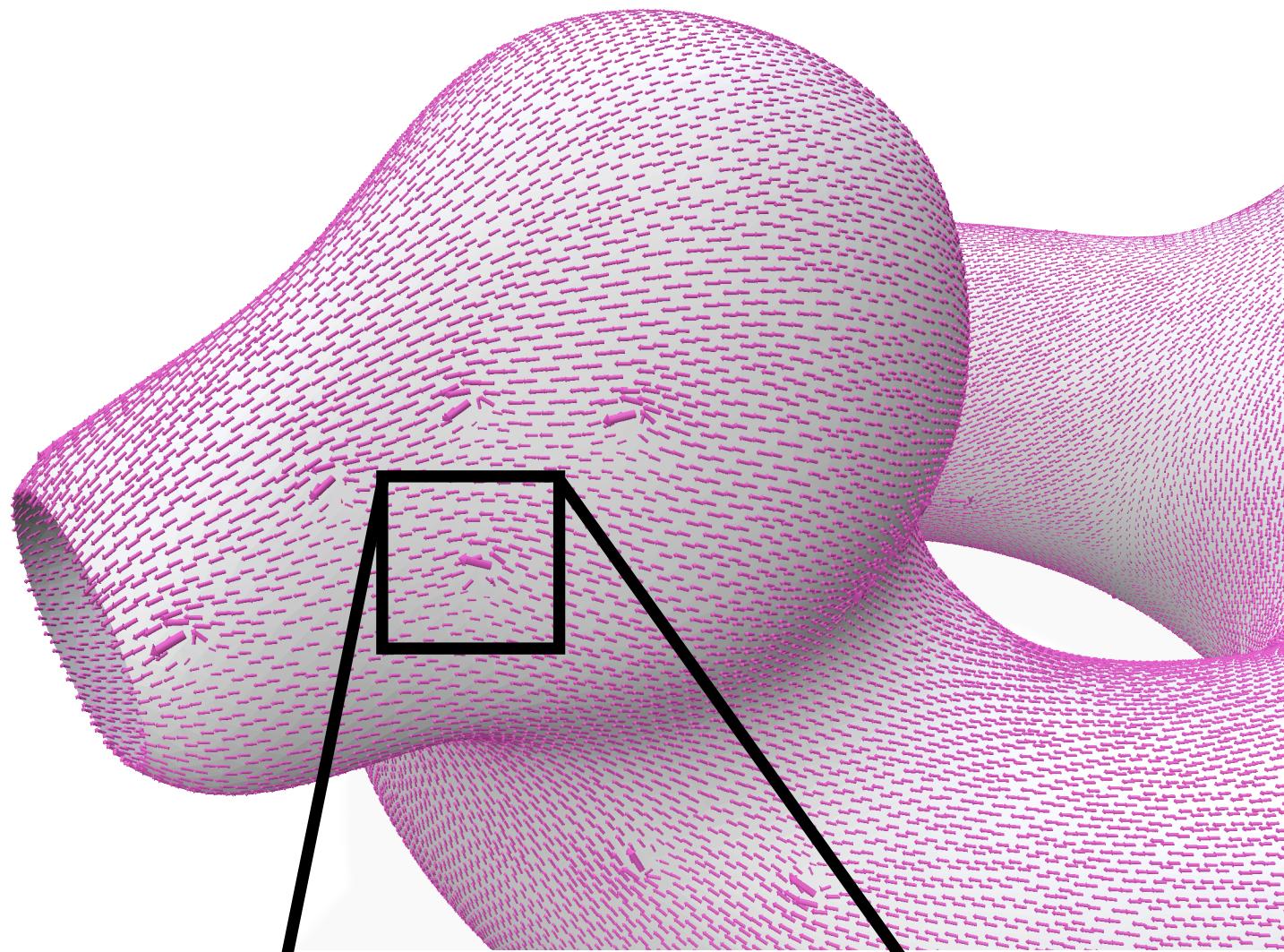
- ▶ **Fast** (greedy strategy)
- ▶ **Automatic** singularity placement
- ▶ With **flexible** user control



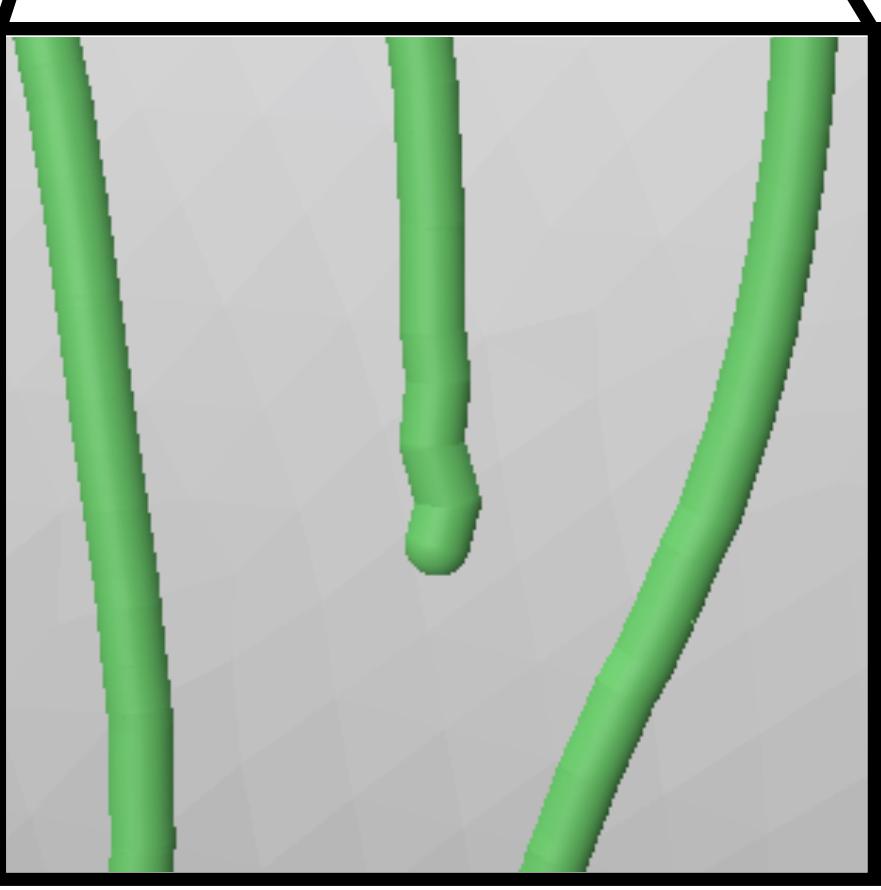
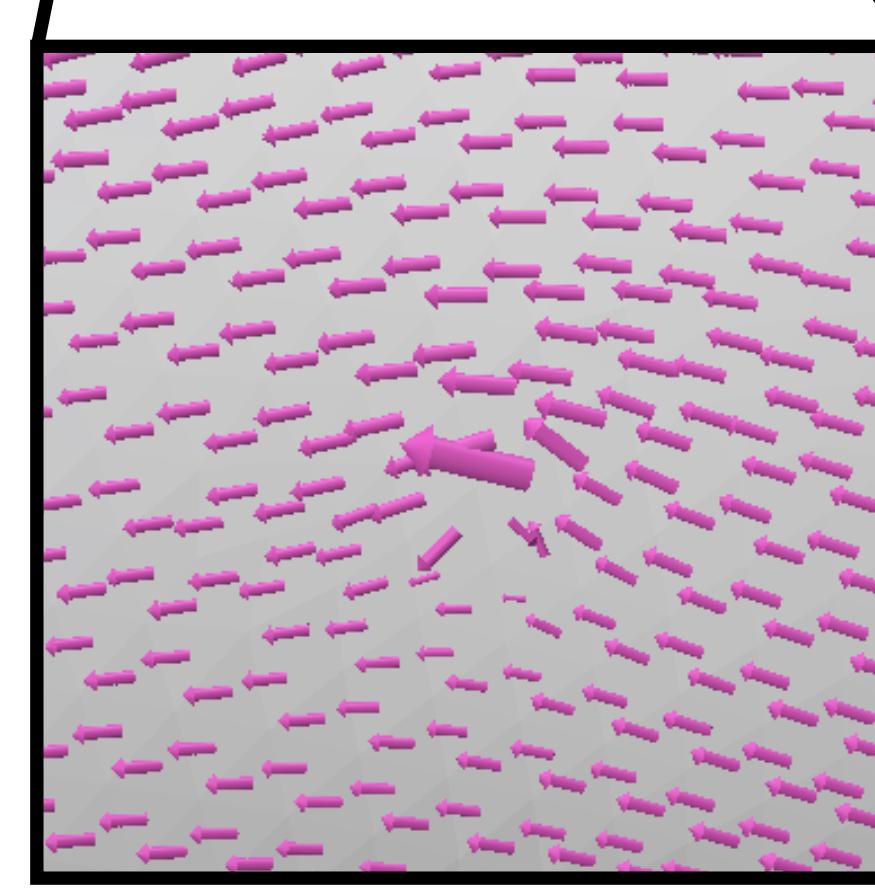
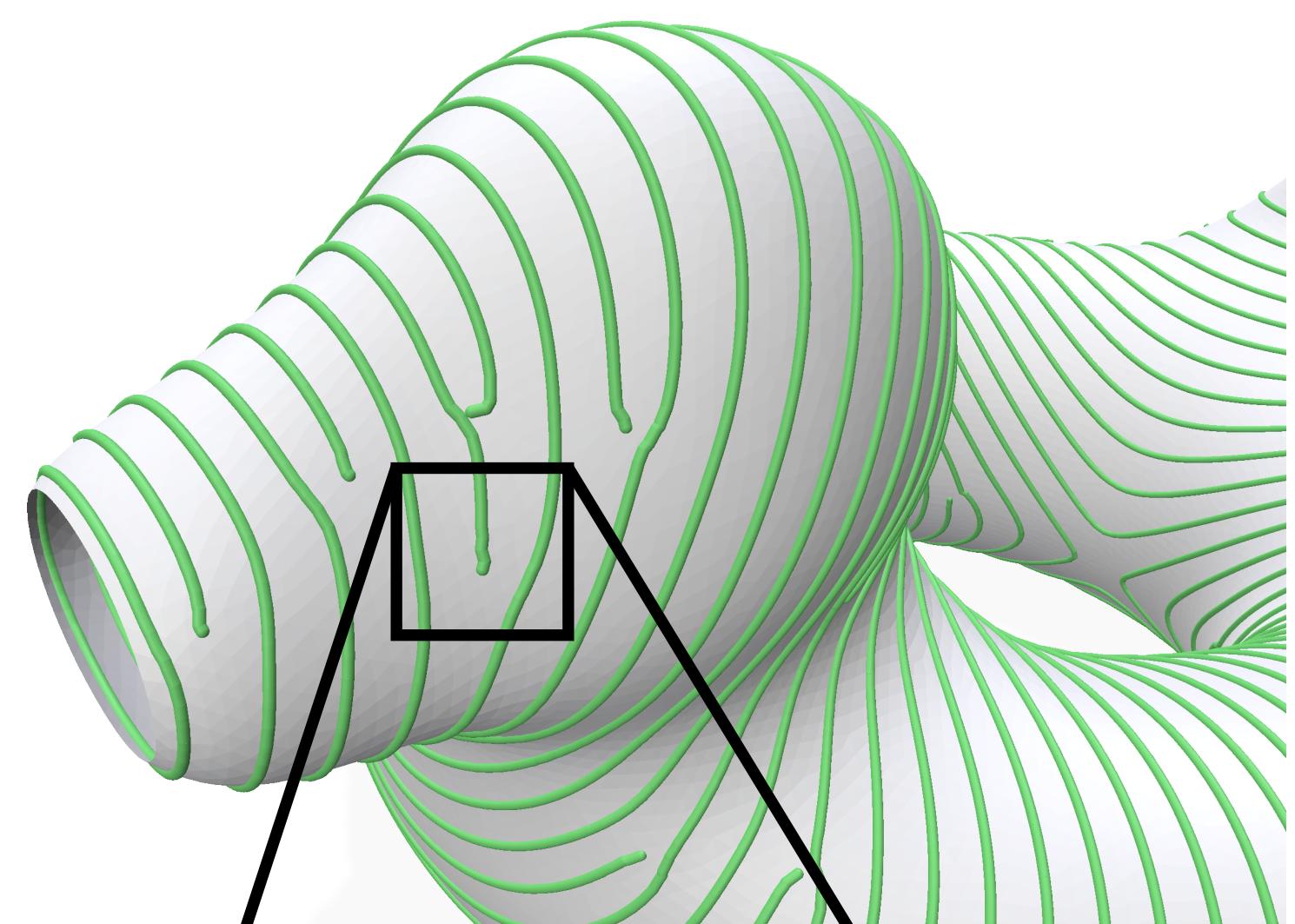


# Computing stripe patterns

Singular vector fields



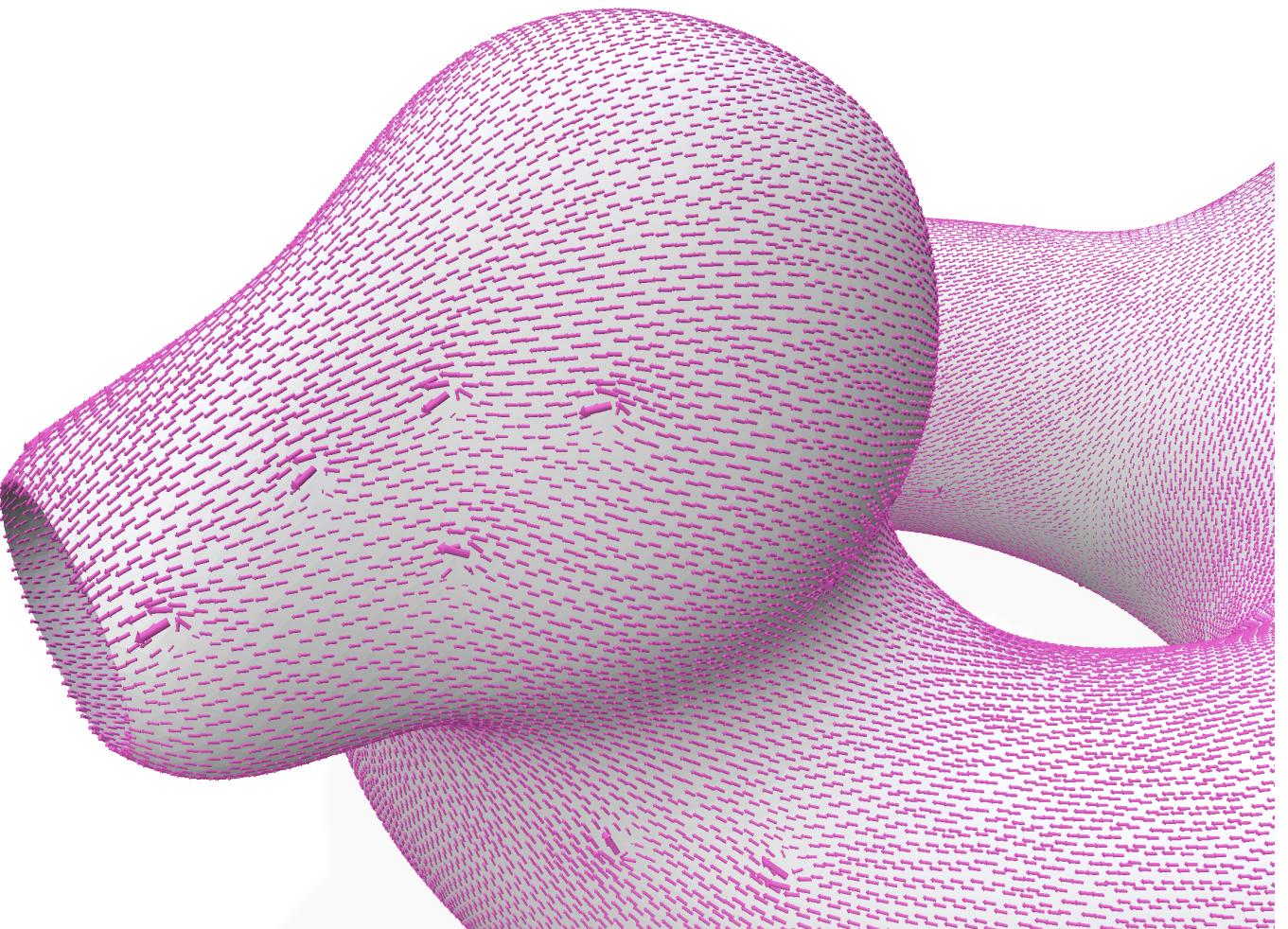
Stripes



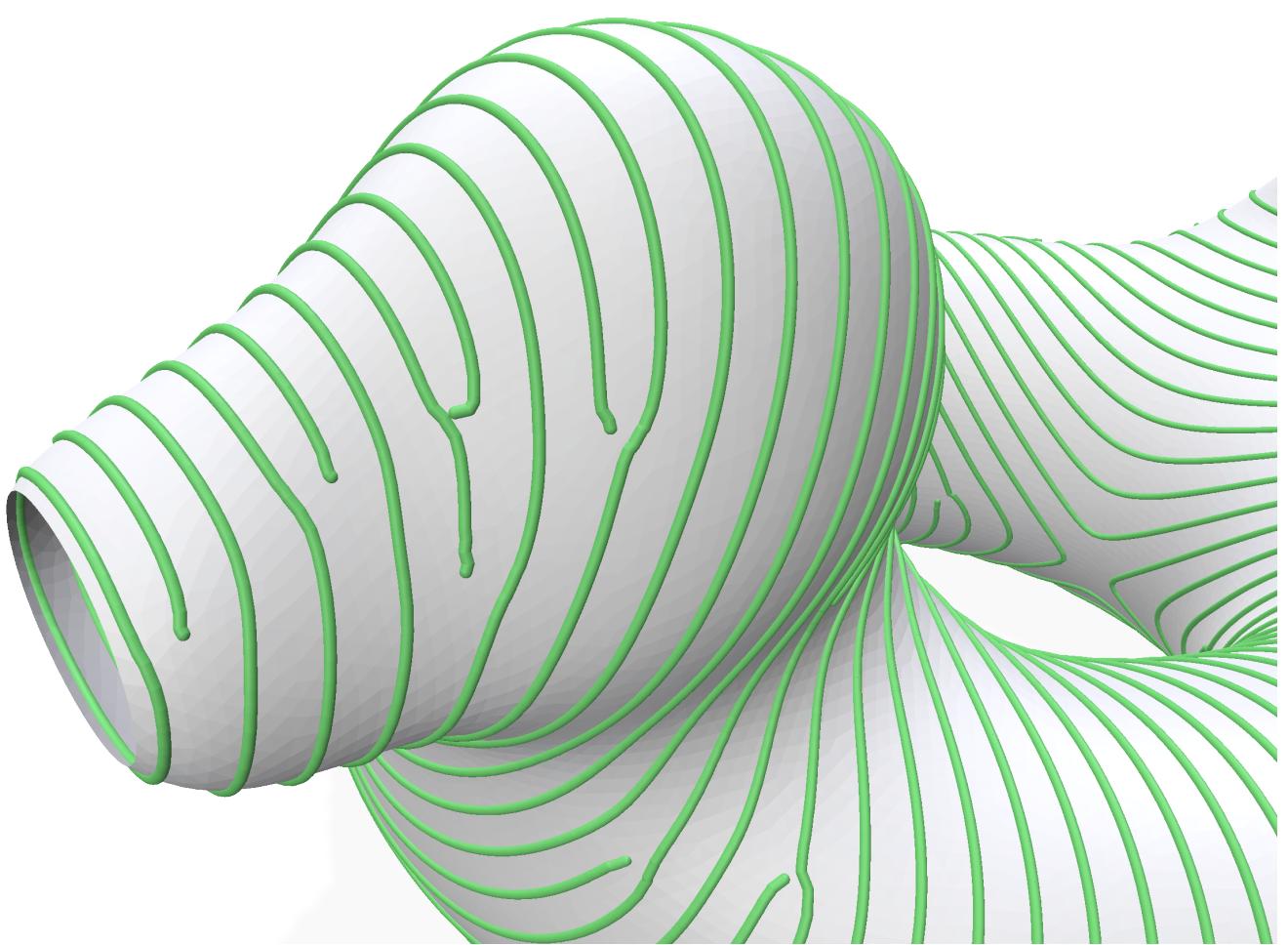
# Computing stripe patterns

Singular vector fields

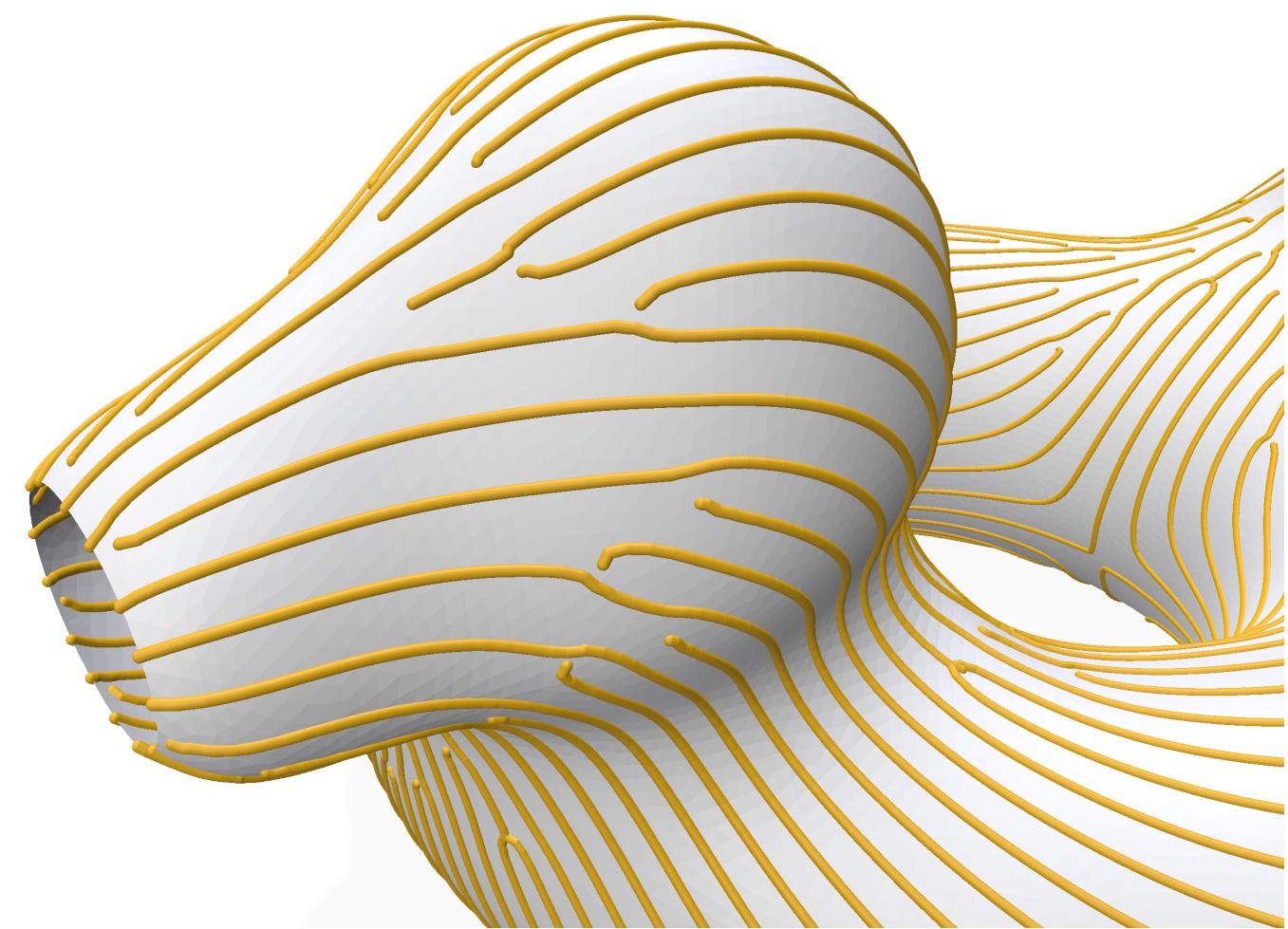
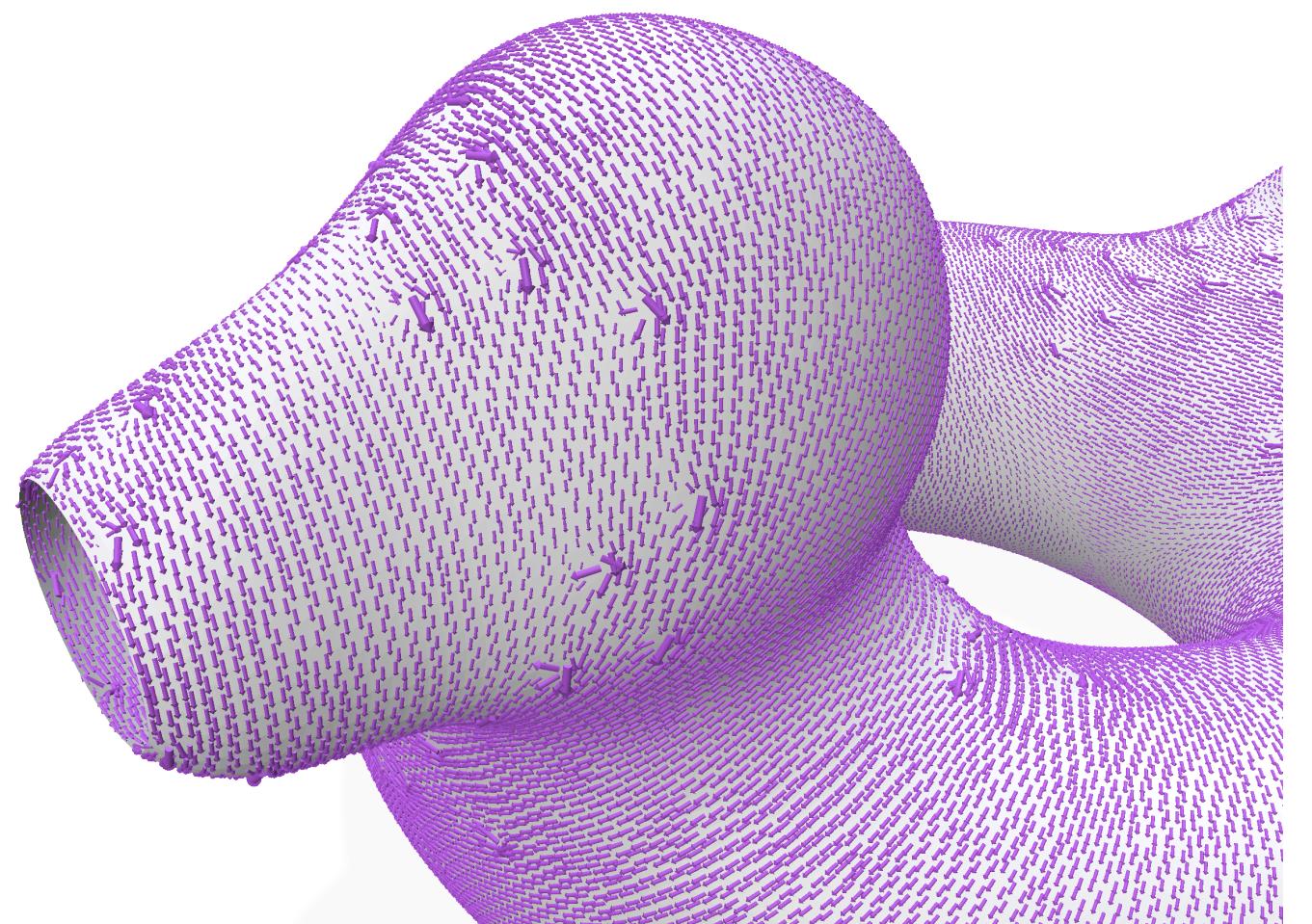
Course



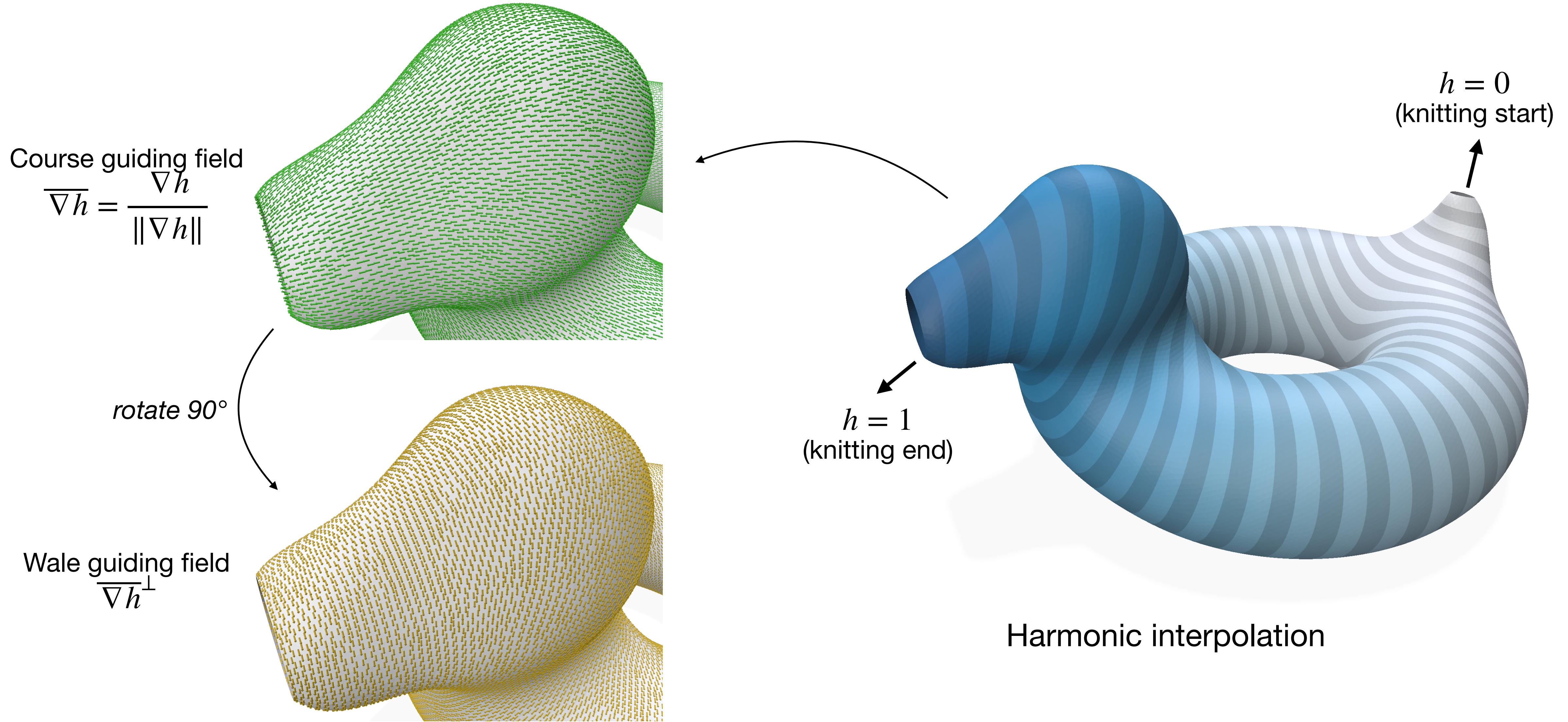
Stripes



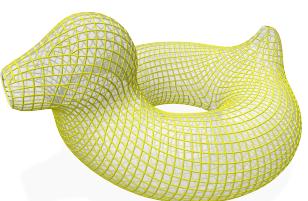
Wale



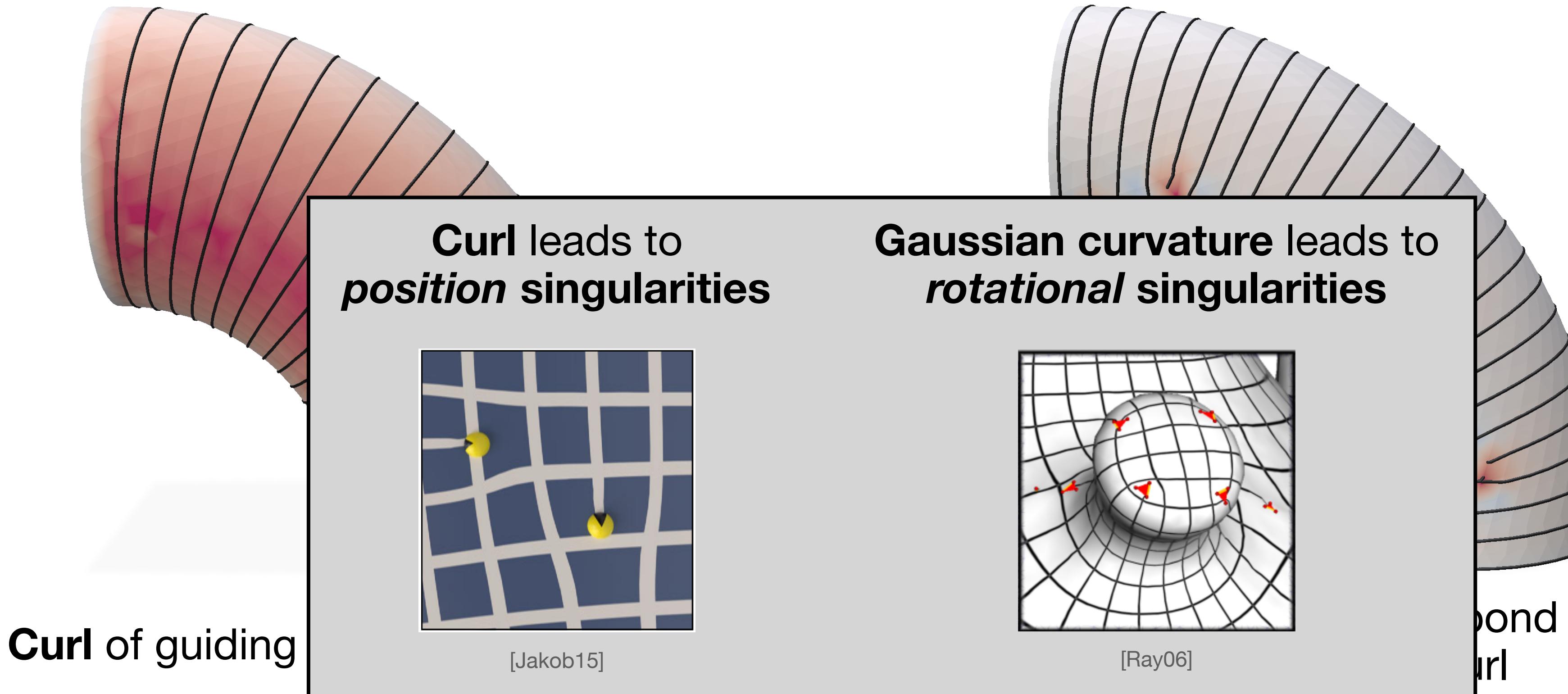
# Knitting is guided by a *time function*



Vector fields are normalized to promote **knit graph regularity**



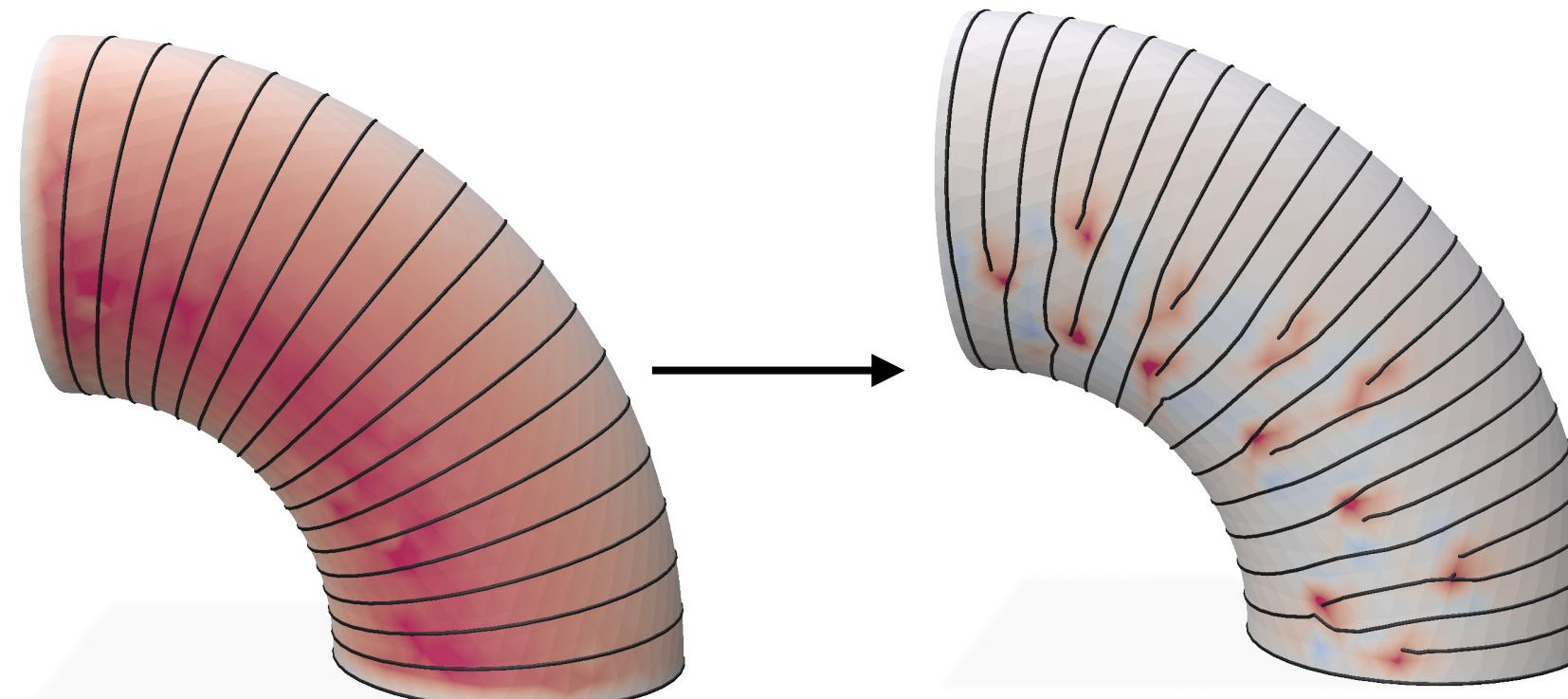
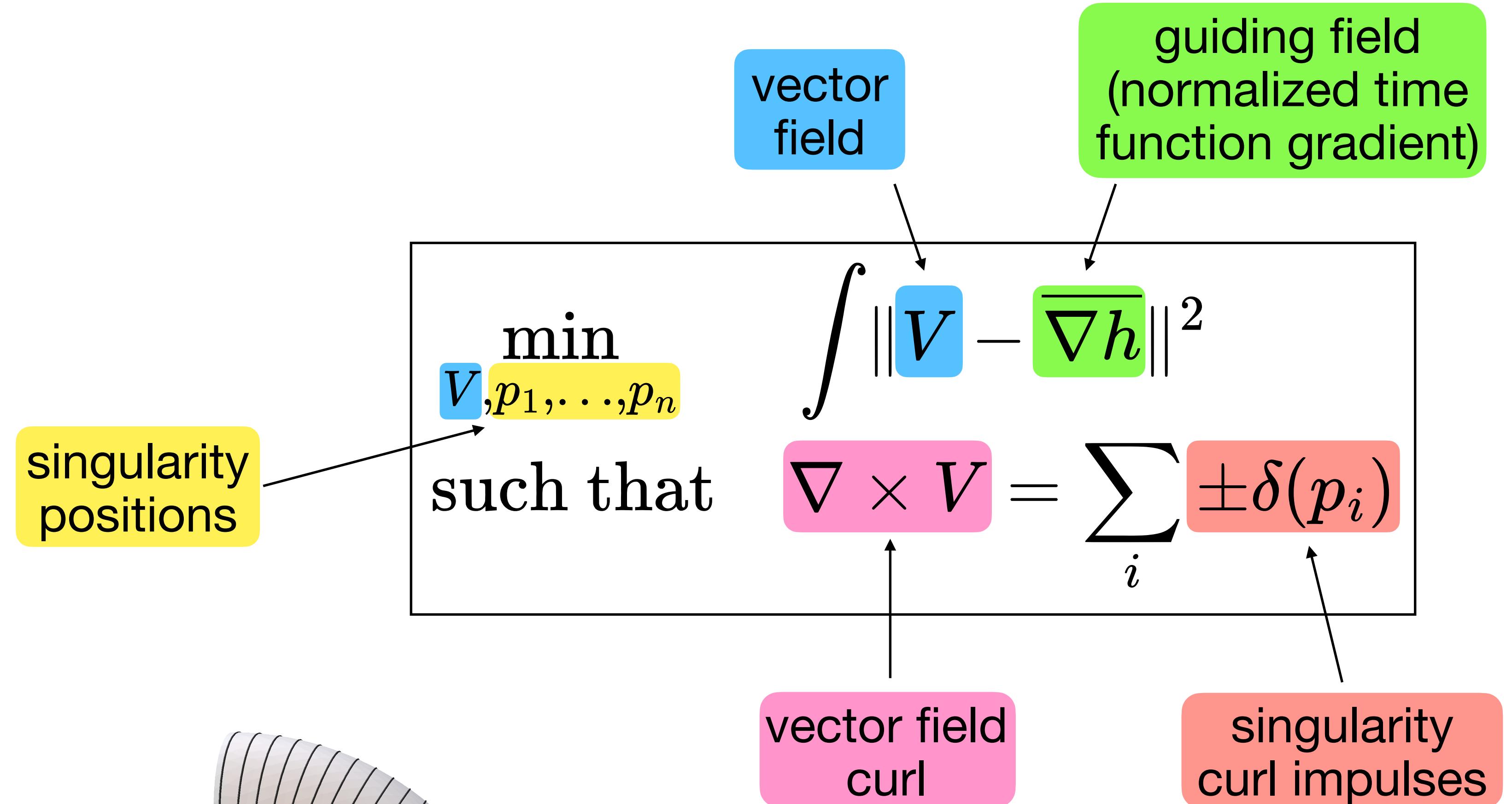
# Curl signal

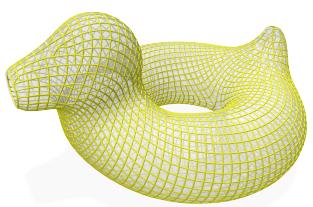


- ▶ Curl indicates where level set spacing **changes quickly**
- ▶ Inserting **singularities** allows for more **even spacing**
- ▶ Idea: Quantize curl to a discrete set of singularities, and use curl of  $\nabla h$  as a heuristic for placement!



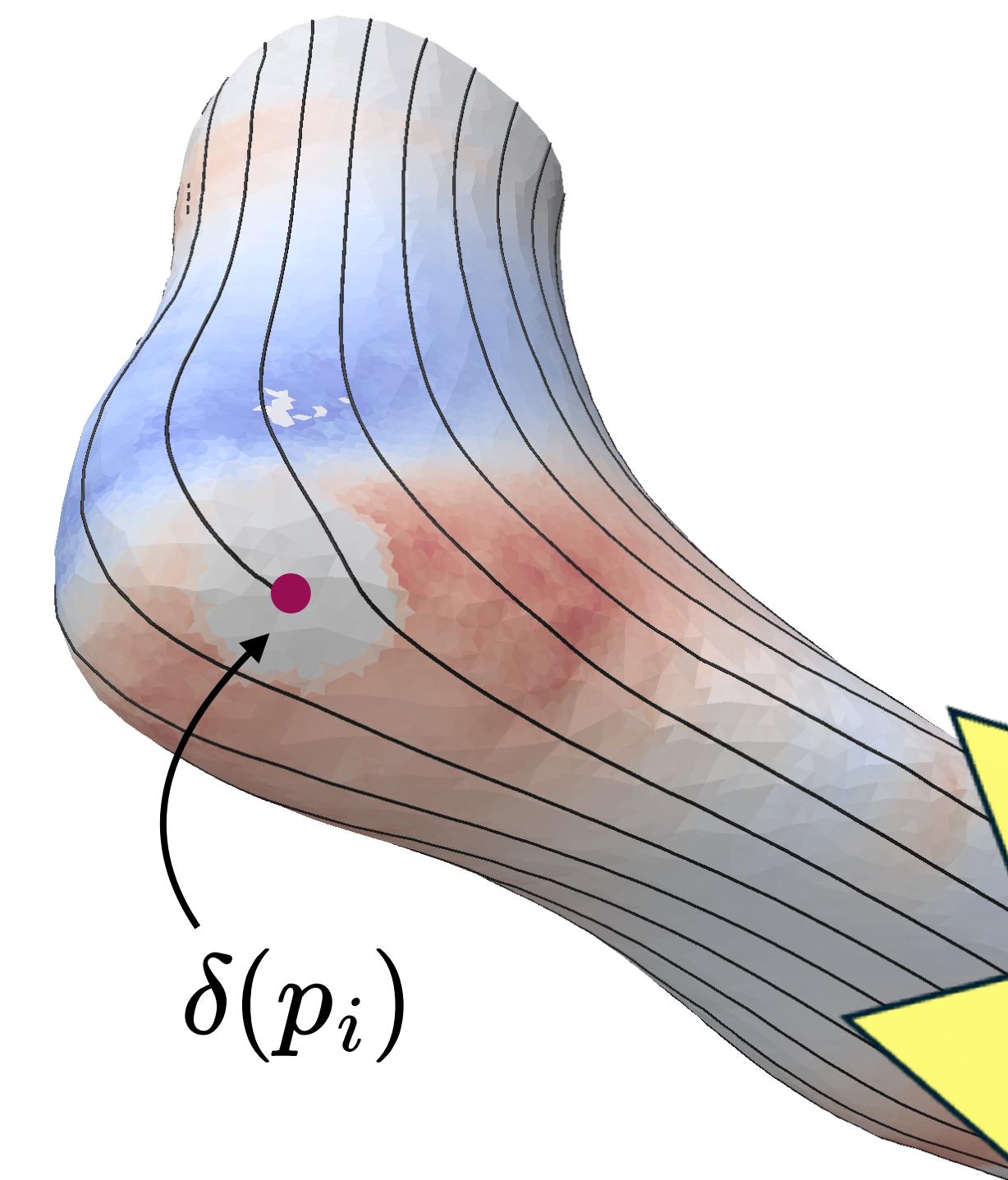
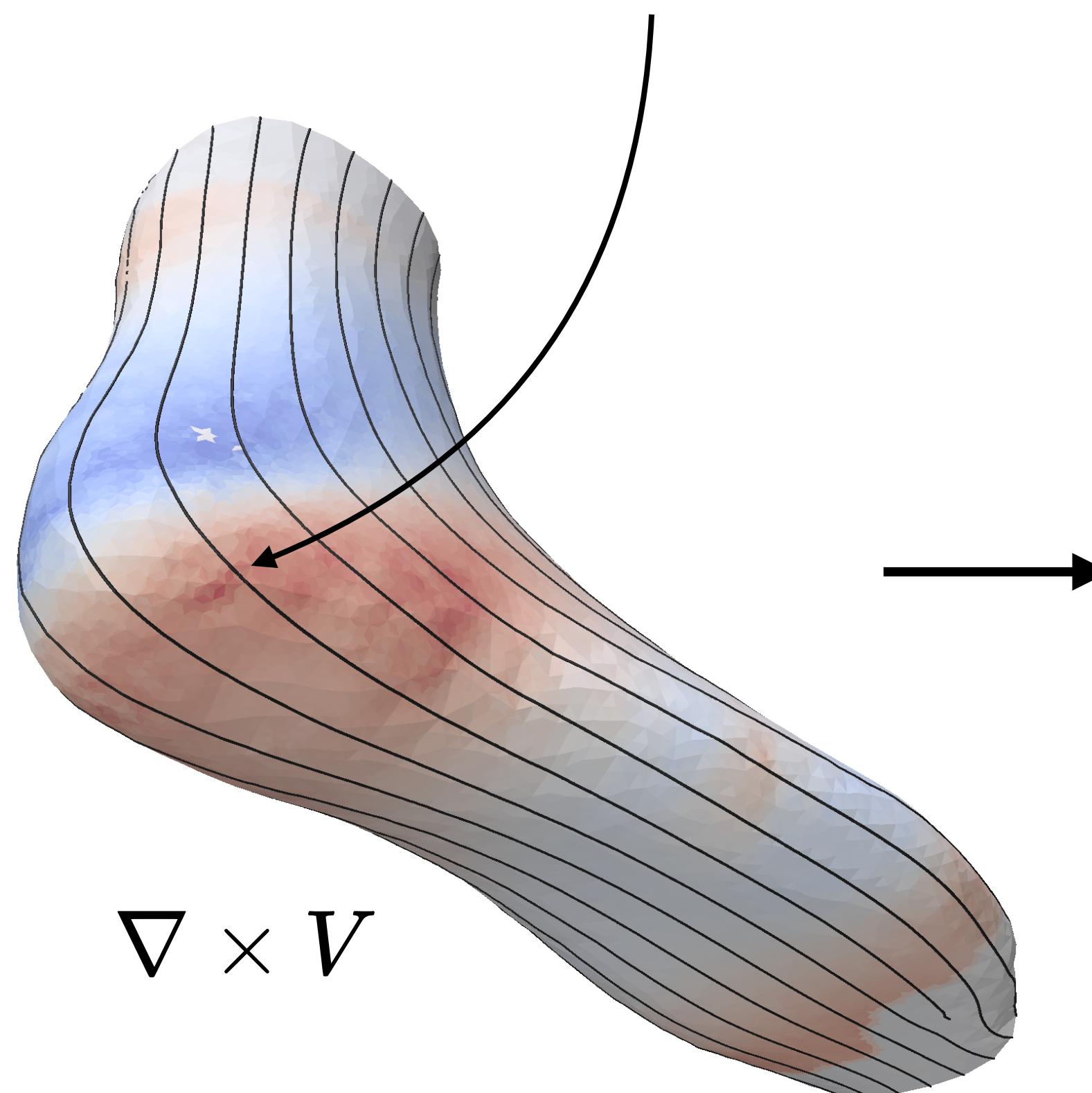
# Curl quantization





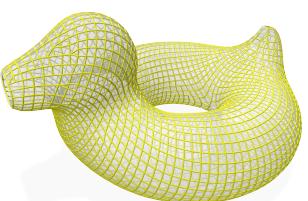
# Singularity placement strategy

Insert singularity at point  
of max absolute curl

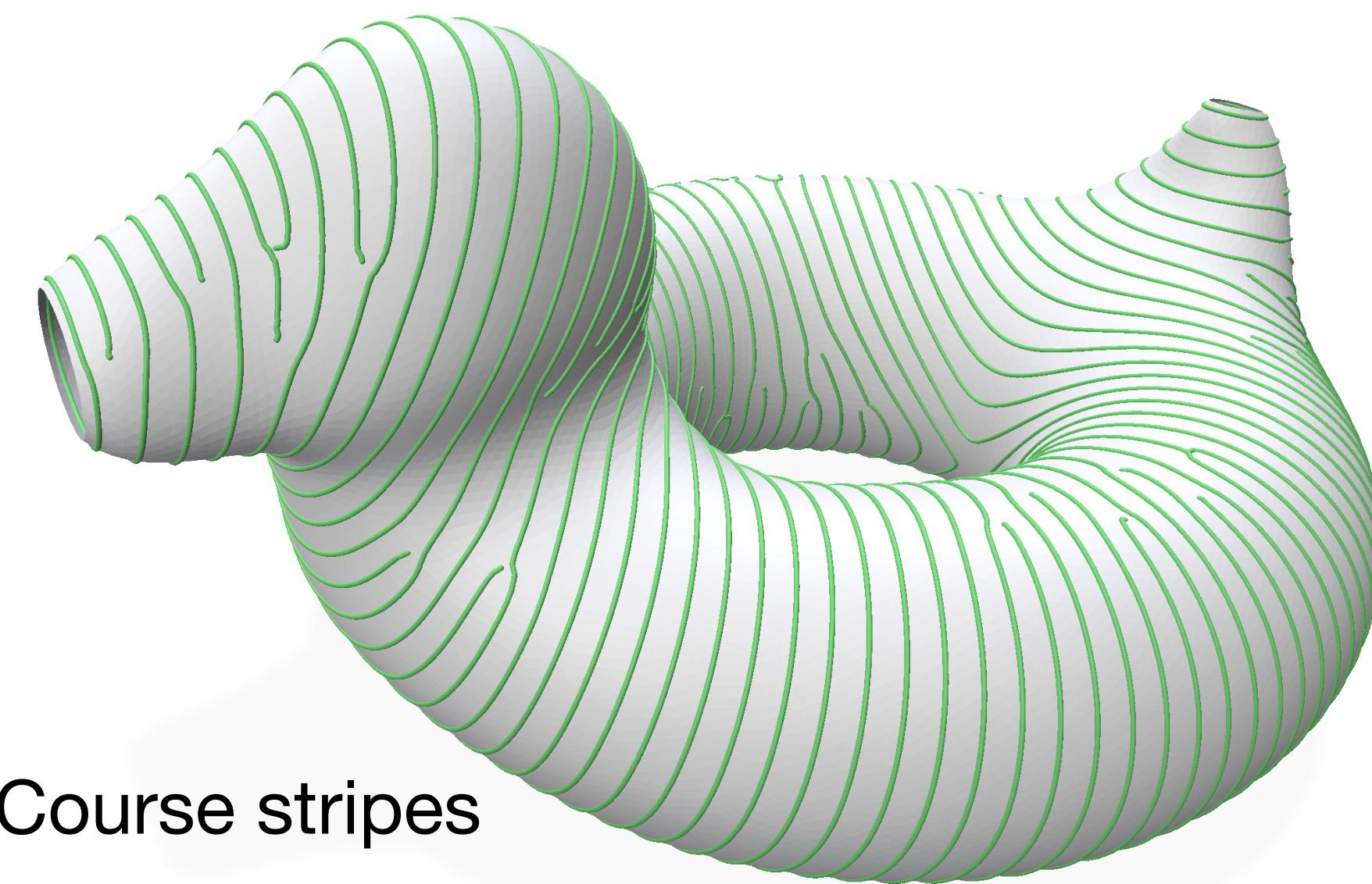


Repeat until objective  
 $\int \|V - \overline{\nabla h}\|^2$   
stops decreasing

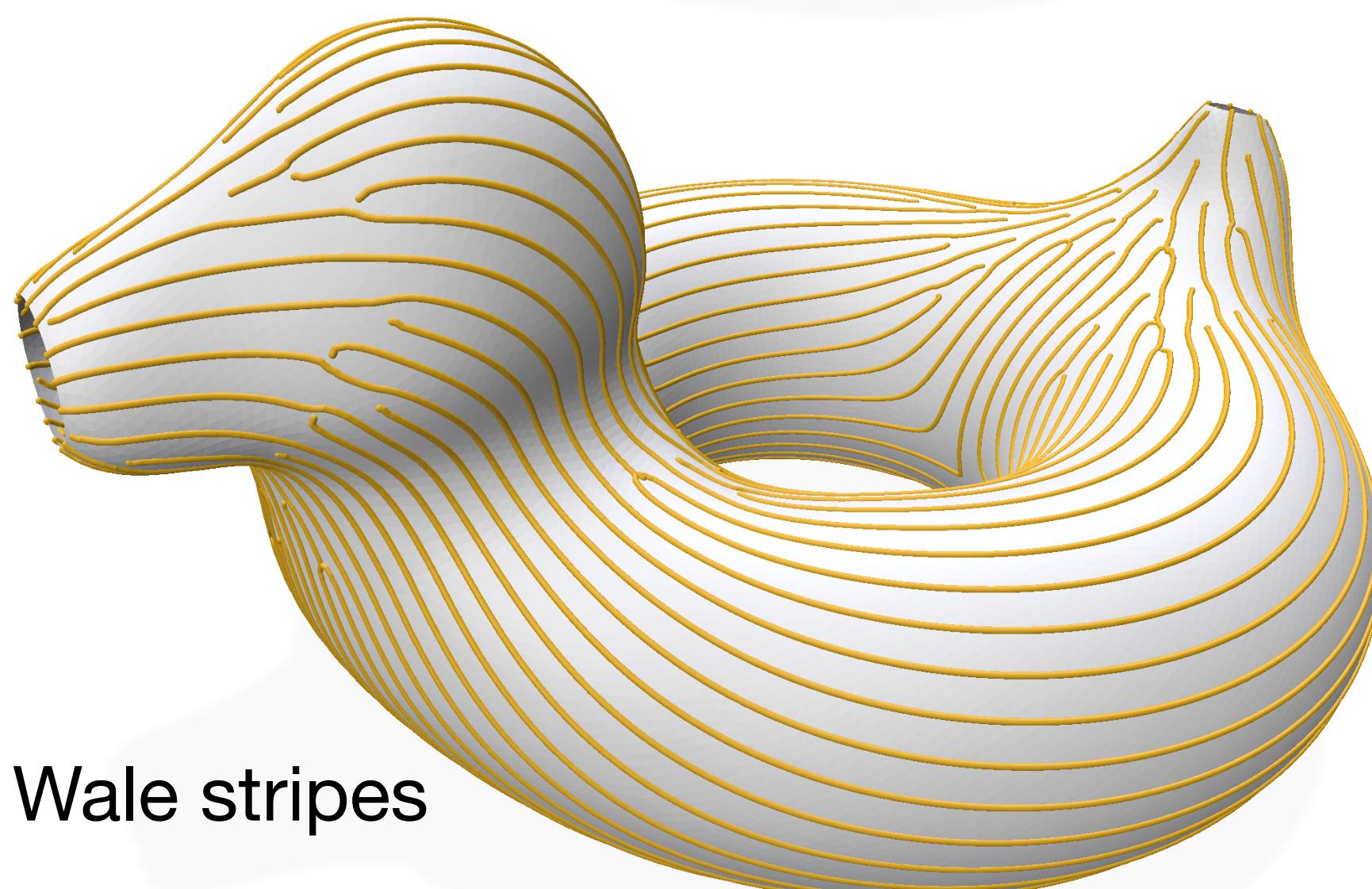
Course stripes  
guaranteed helix-free  
via [Mitra24]!



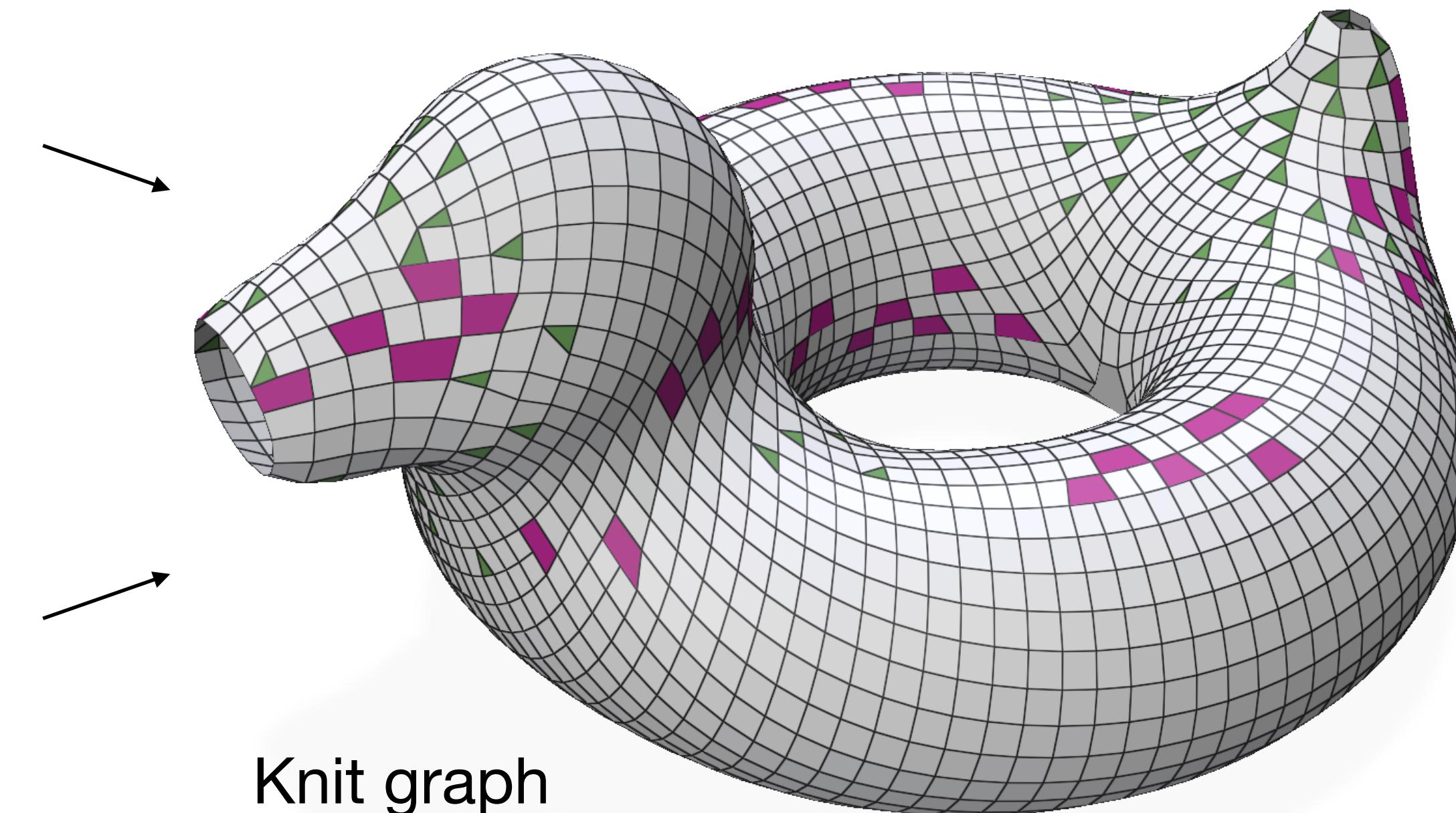
# Knit graph construction



Course stripes



Wale stripes



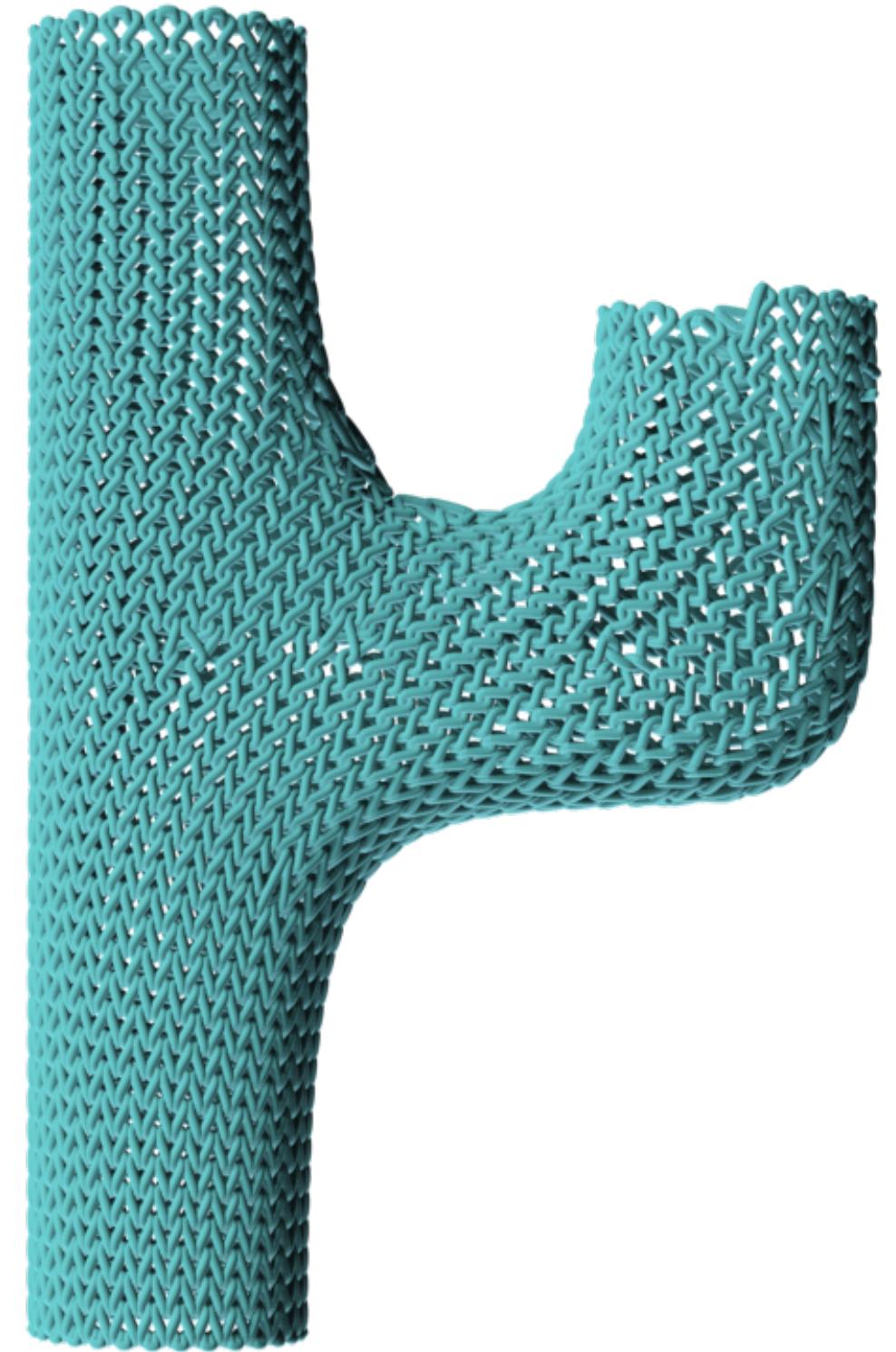
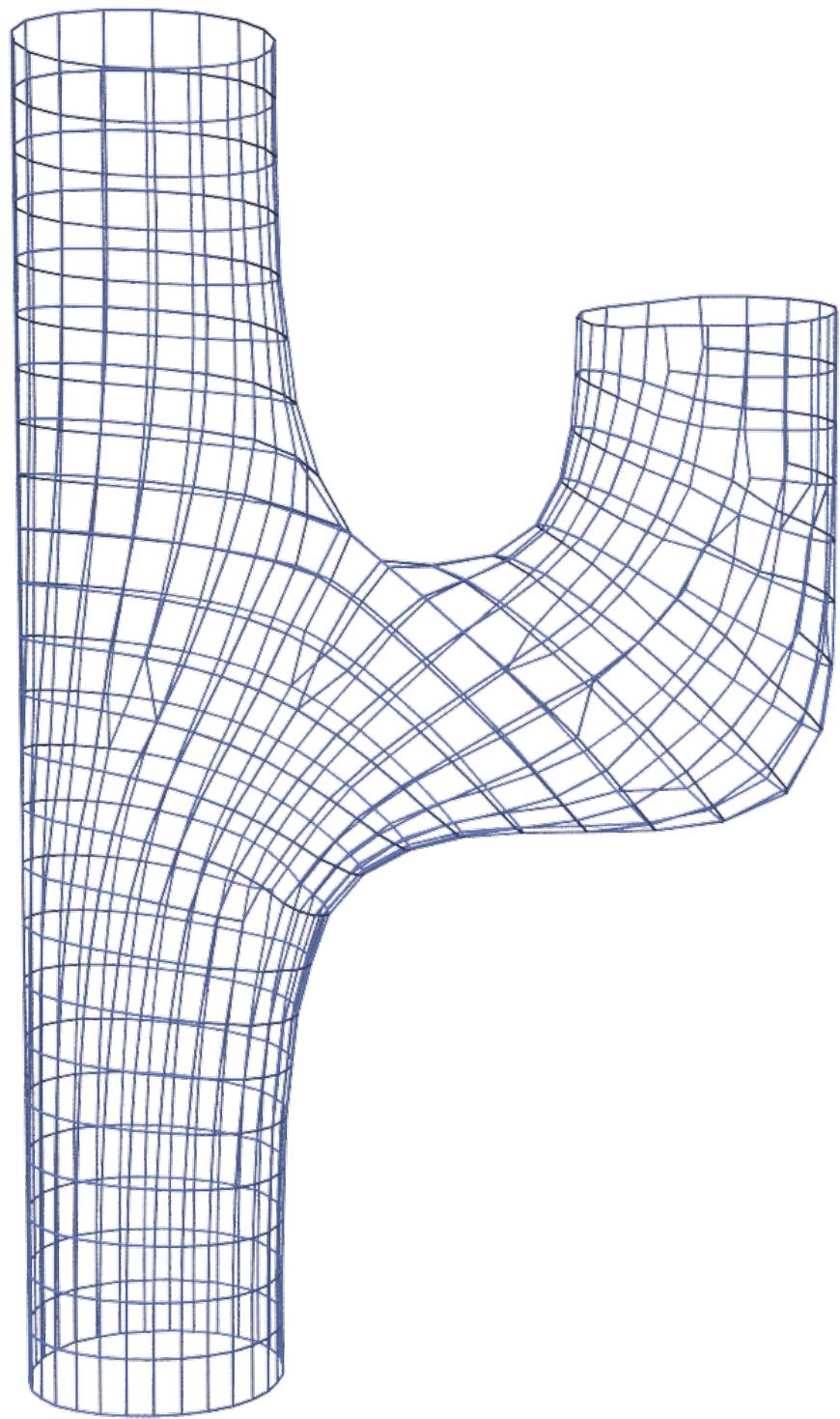
Knit graph

**Pentagons** = short row ends

**Triangles** = wale increases and decreases



# Results



compared to  
[Mitra23]



Model	#V	#F	Time	Speedup
Sock	279	538	< 1 s	53x
Glove	37.5k	74.5k	56 s	timeout
Cactus	391	736	< 1 s	100x
Dress	8.6k	17k	11.5 s	31x
4-panel skirt	1.1k	1.9k	< 1 s	no cut&sew
Pants	1.2k	2k	< 1 s	no cut&sew

## Knit graphs

- ▶ Machine-knittable

## Yarn-level renders

- ▶ Renders are smooth without any yarn relaxation!

## Machine knits

- ▶ Scheduling done with AutoKnit

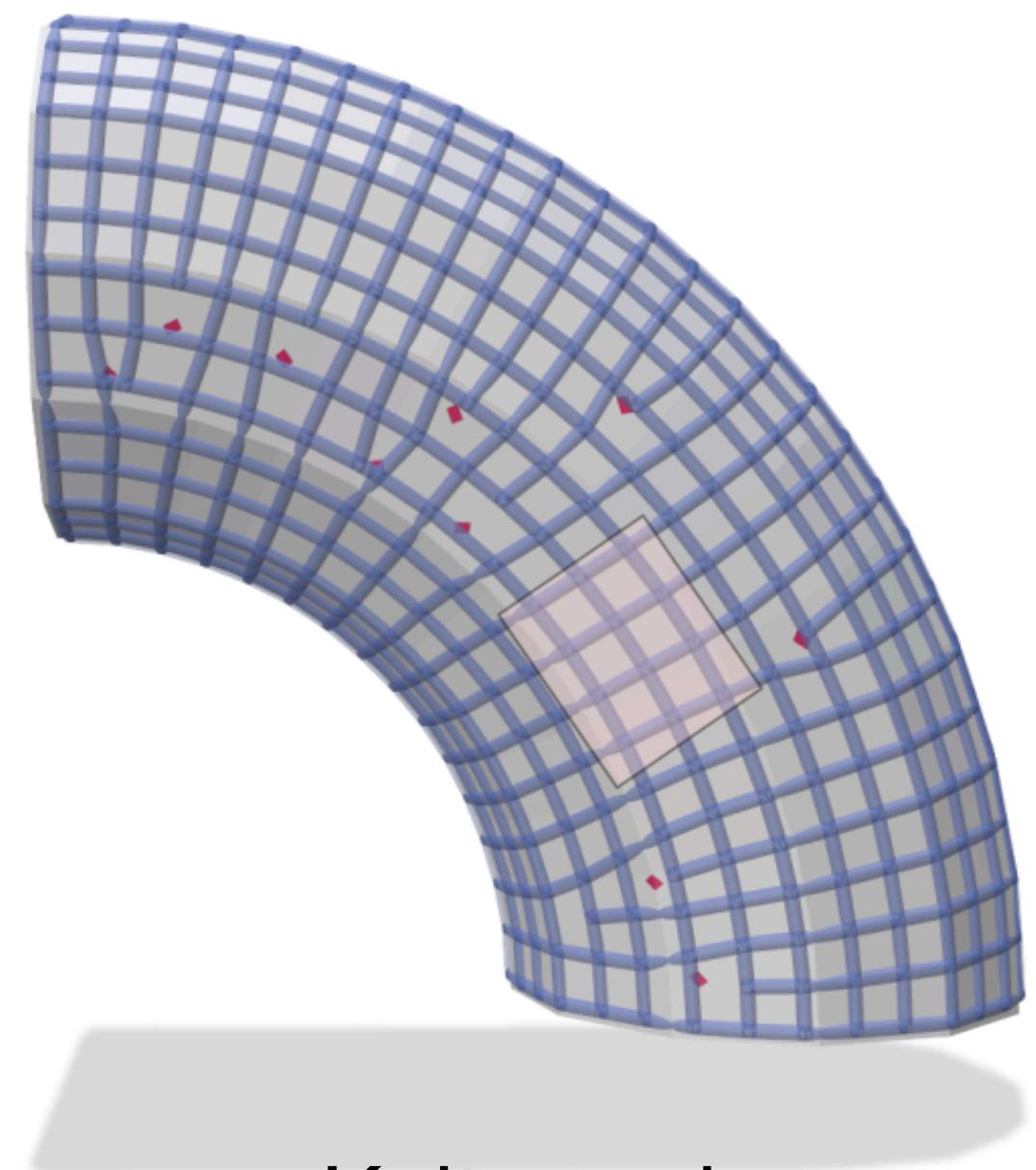
# User has control over singularity placement

through curl signal editing

**Prevent singularities** in specified region  
by **masking** the curl signal

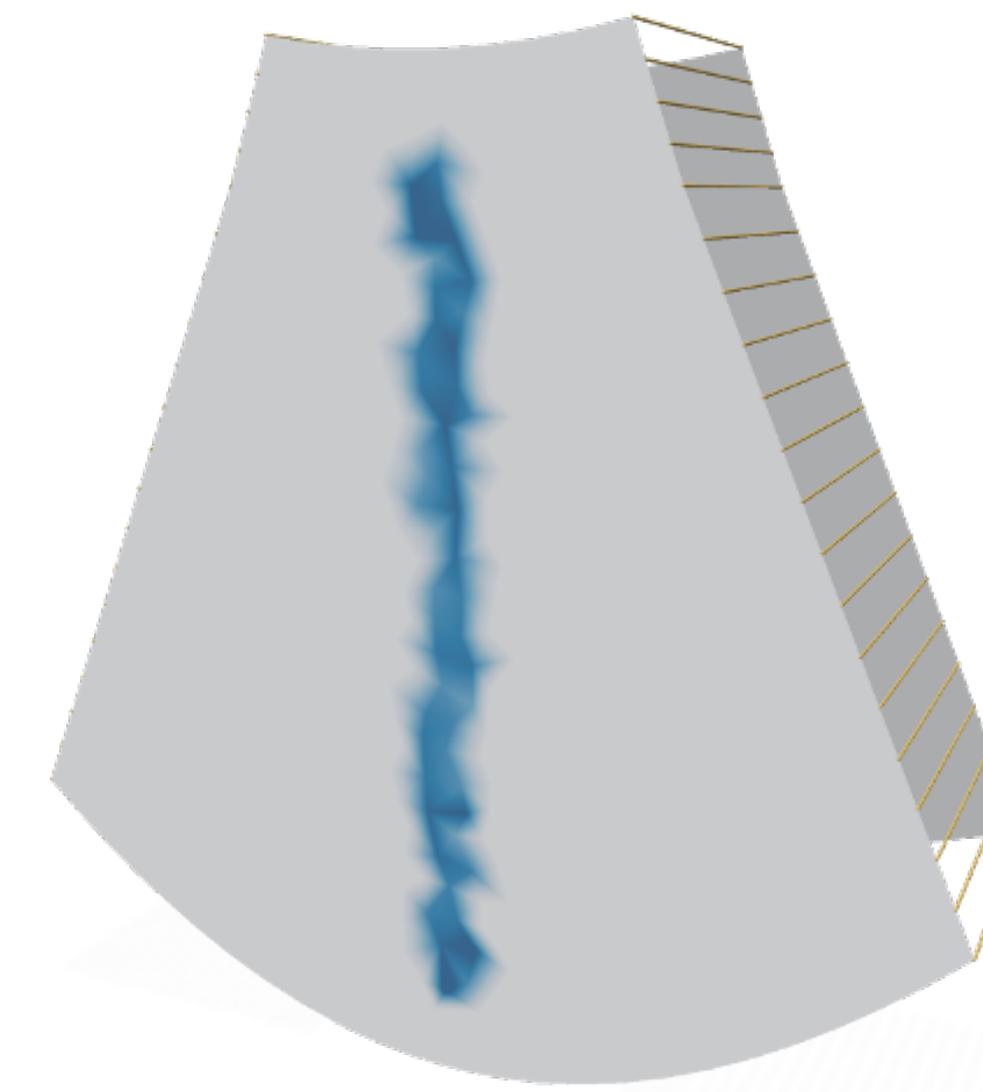


Curl signal

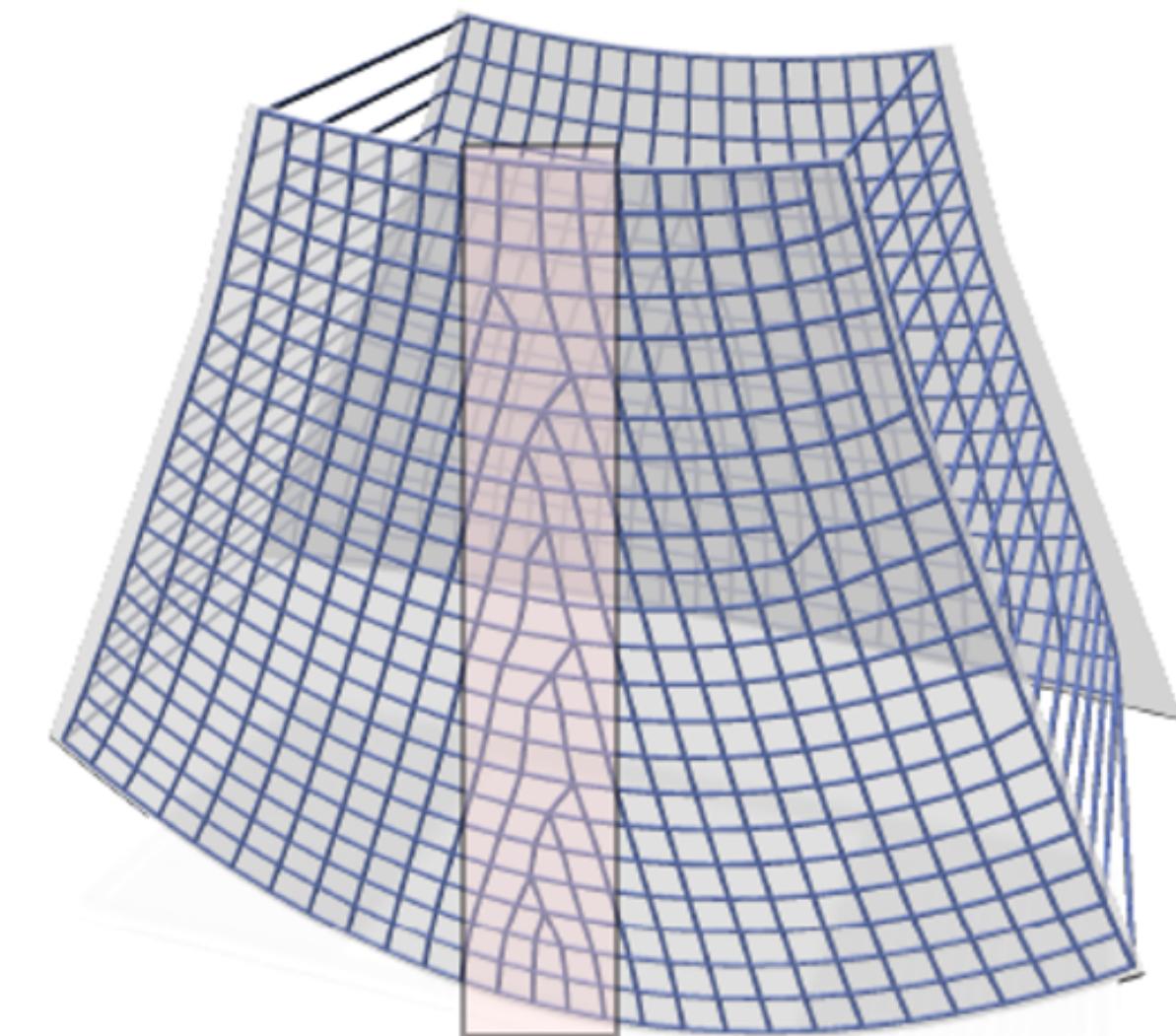


Knit graph

Create **seam** in specified region  
by **boosting** the curl signal

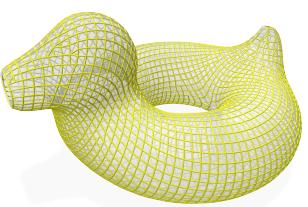


Curl signal



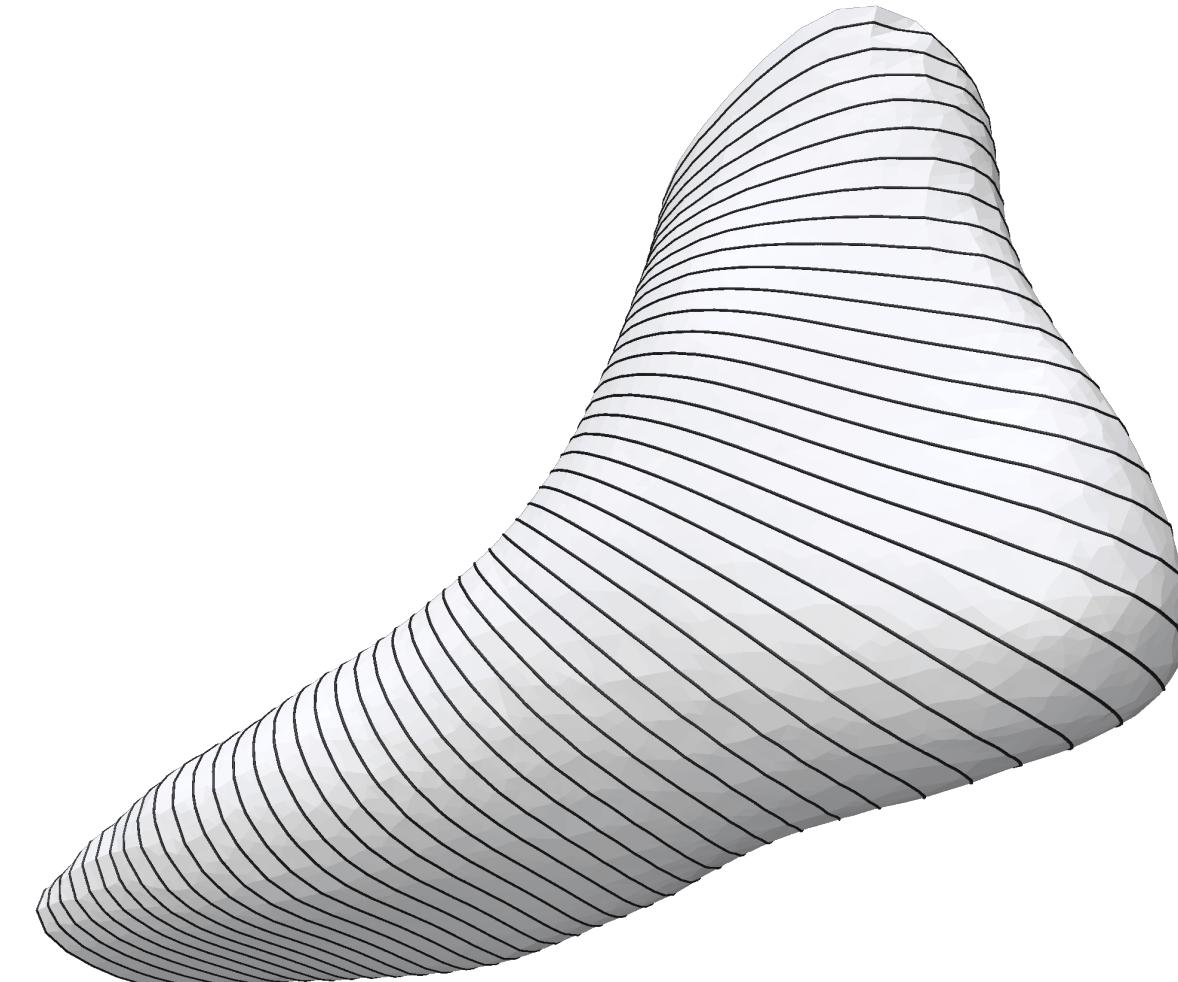
Knit graph

Not possible in previous works!

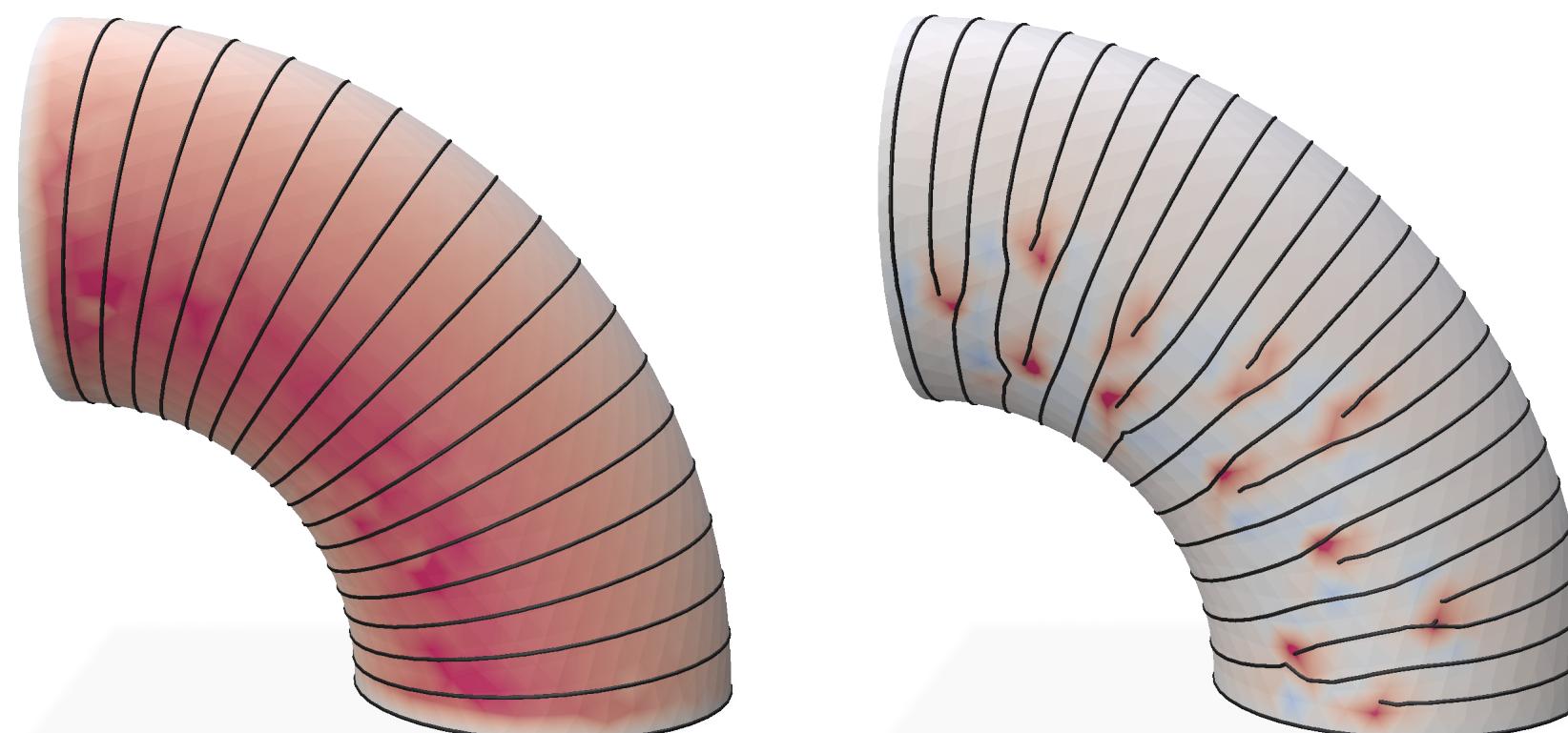


# Conclusion

Fast, Automatic, Flexible  
placement of knit singularities



Key idea:  
Curl indicates where to place singularities



Coming up:

- ▶ Global strategy: place all singularities in one optimization
- ▶ Design better guiding fields, or optimize guiding field and knit graph jointly

