

Automatic isobath generalisation

By integrating cartographic constraints in a surface-based approach

Willem van Opstal

Martijn Meijers

(1st mentor)

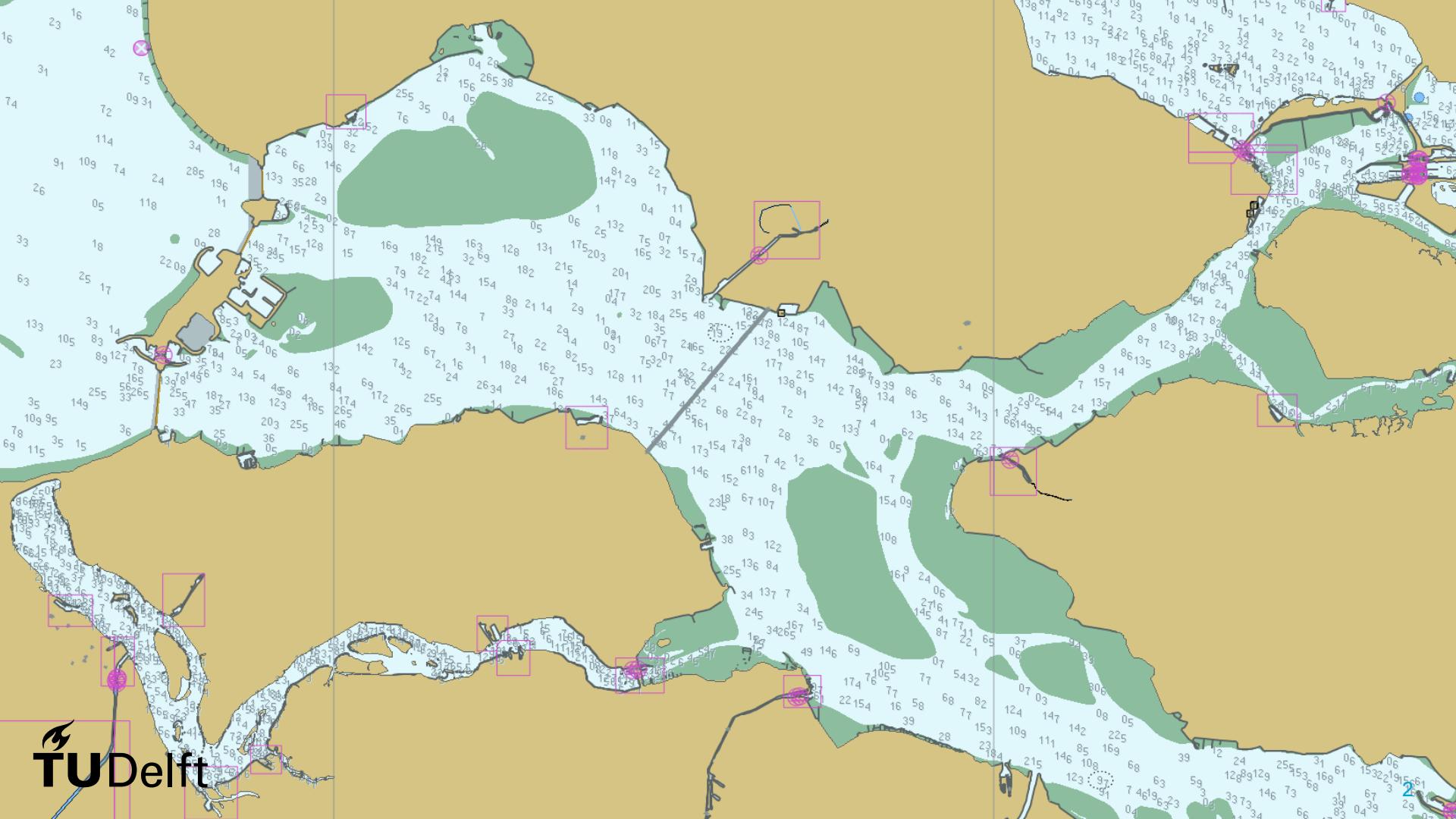
Ravi Peters

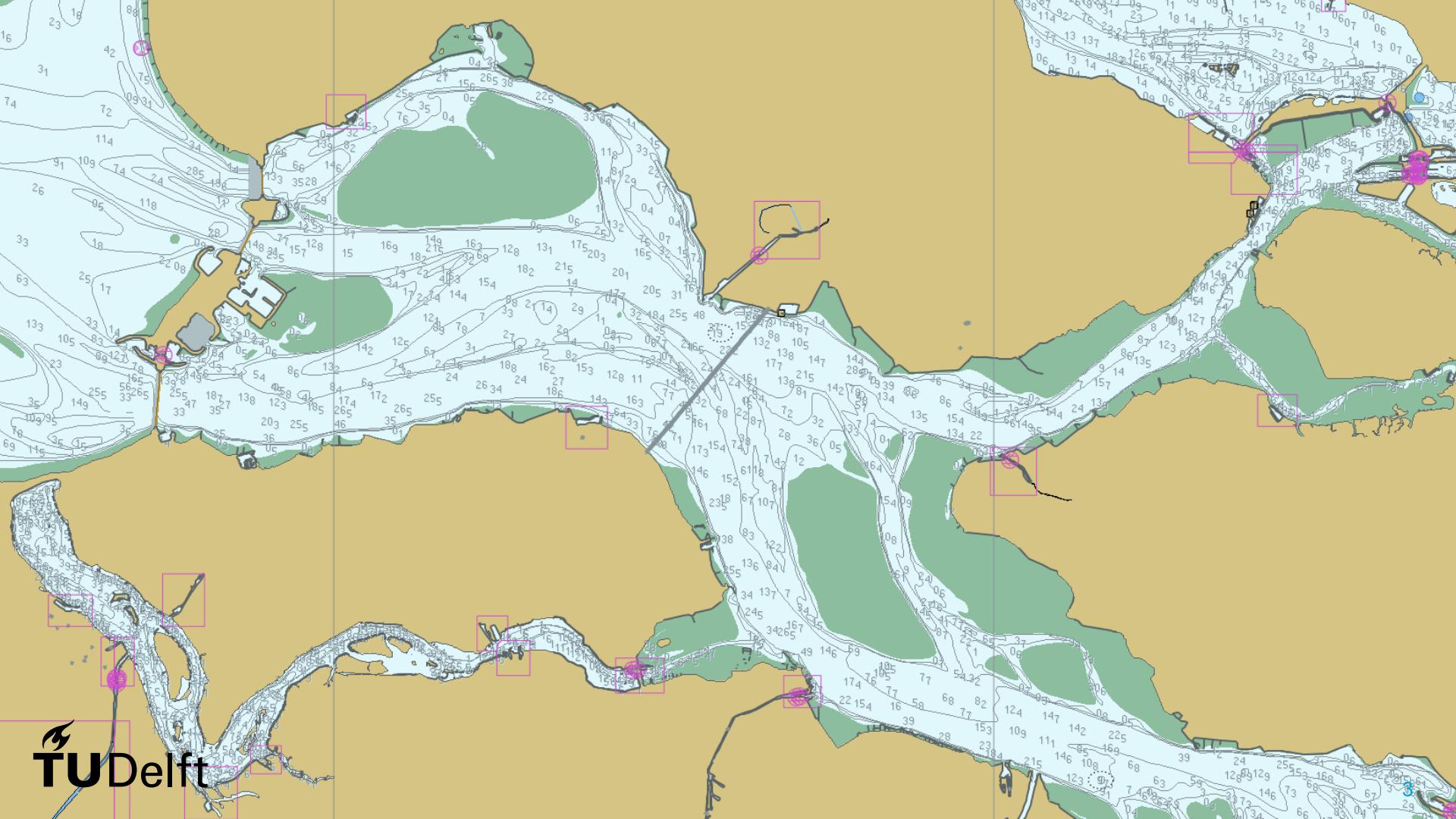
(2nd mentor)

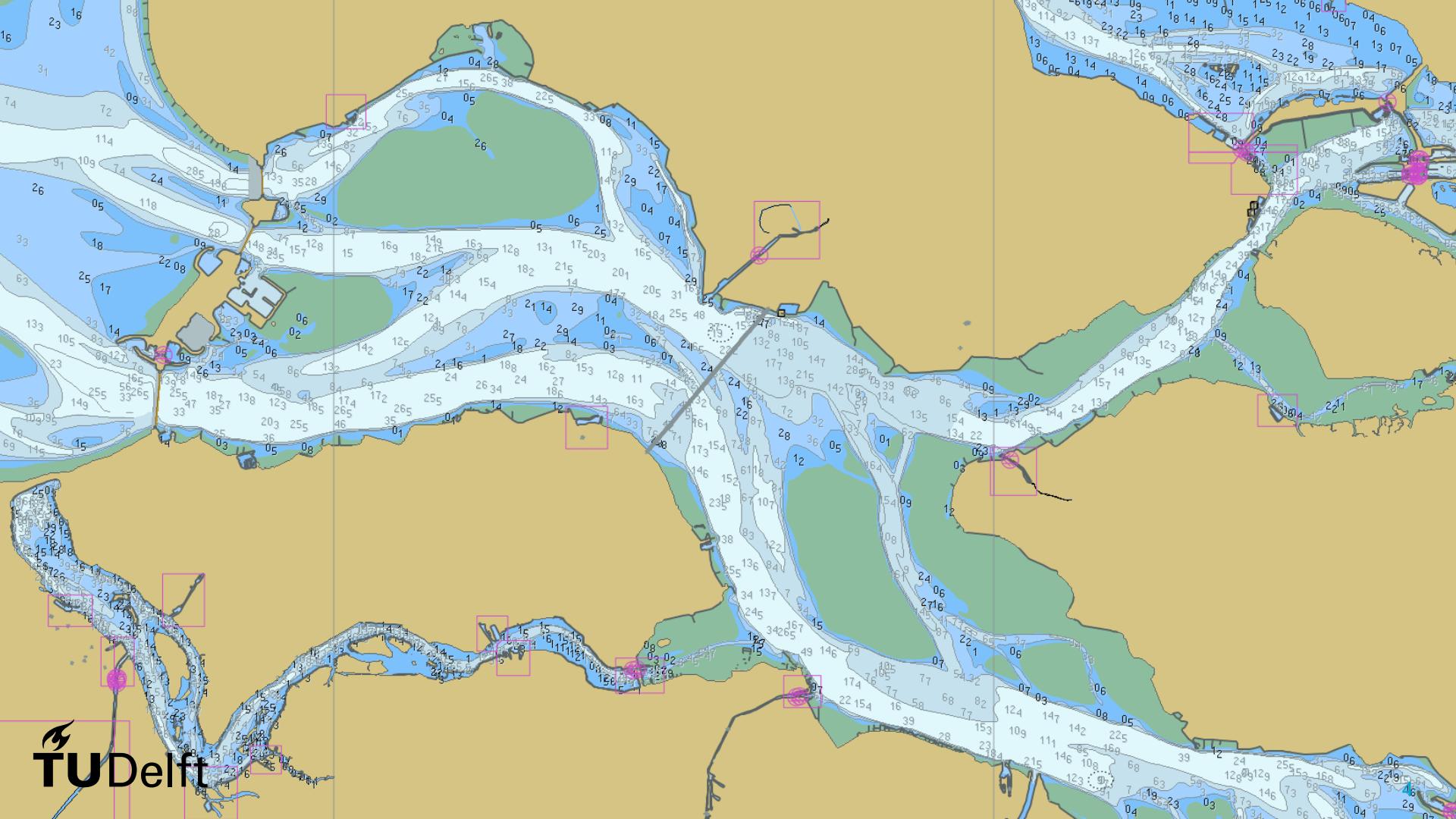
January 16th 2020

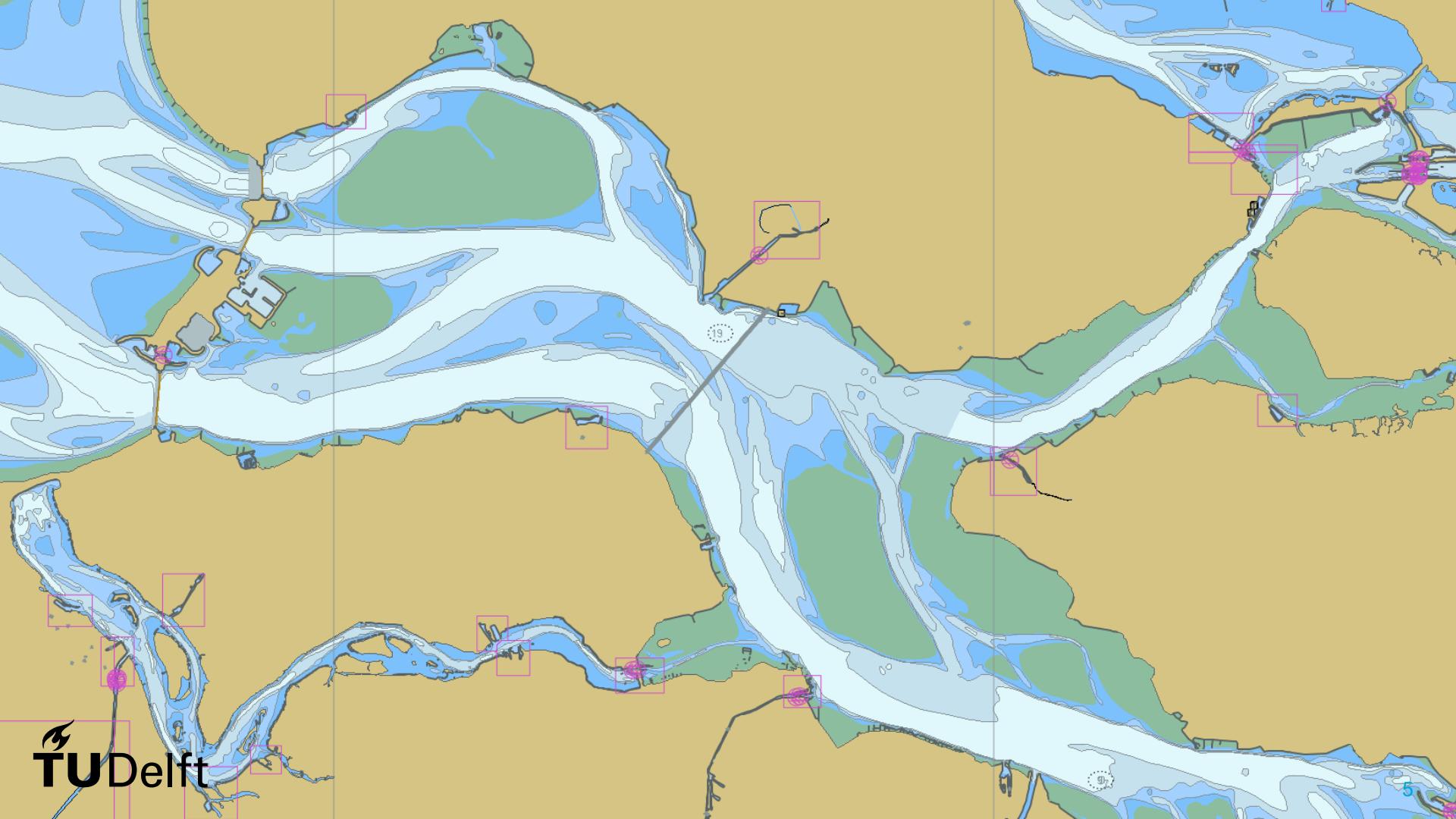
Contents

- Motivation and Problem statement
- Research objectives and Questions
- Related work
- Methodology
- Tools and Data
- Planning







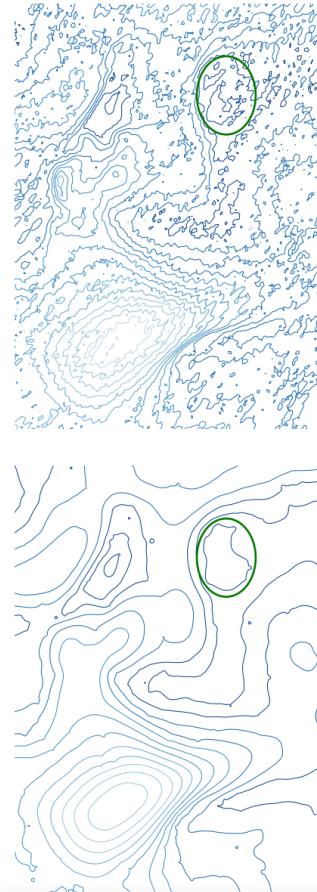


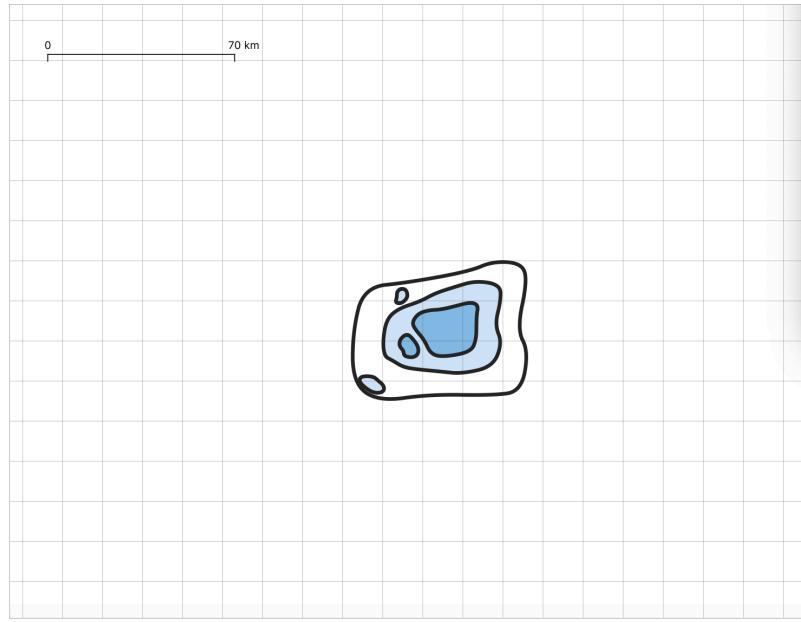
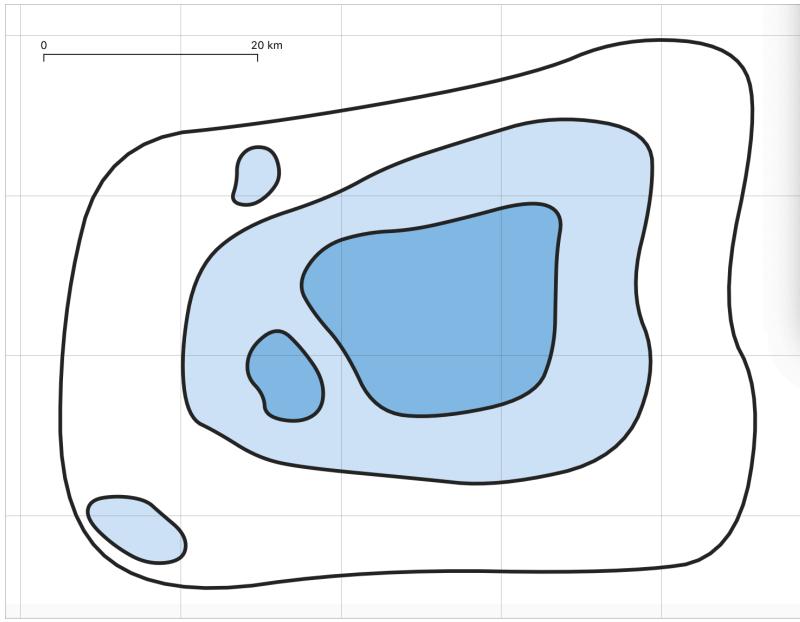
Isobath generalisation

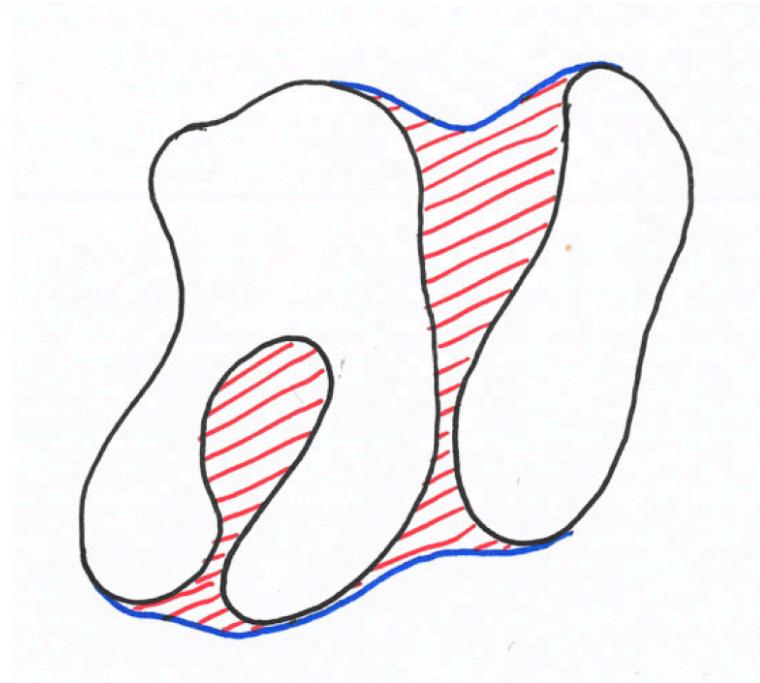
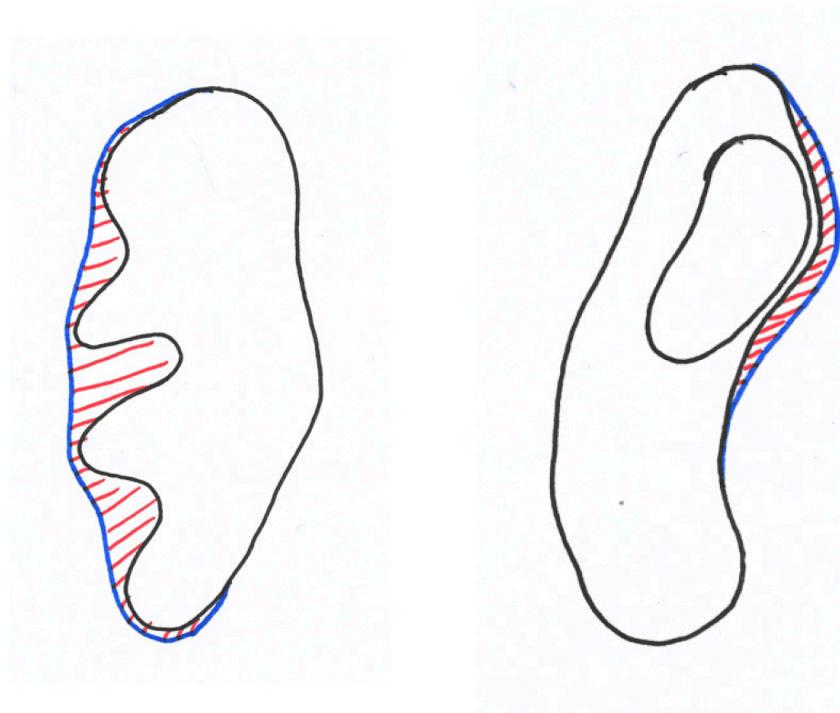
- Cartographic constraints
 - Morphology Seabed shape
 - Legibility Readability
 - Functional Safety
 - Topology Topology
- Currently done manually
- Automation brings:
 - Economic benefits
 - Safety benefits

Problem statement

- Incompatible constraints
 - Chart scales
 - Smoother lines > less morphology
 - Increasing line separation > less morphology
 - Masking safe waters
- Automated process does not exist yet
 - Complex decisions
 - Cartographers insight
 - or Relation with data is destroyed

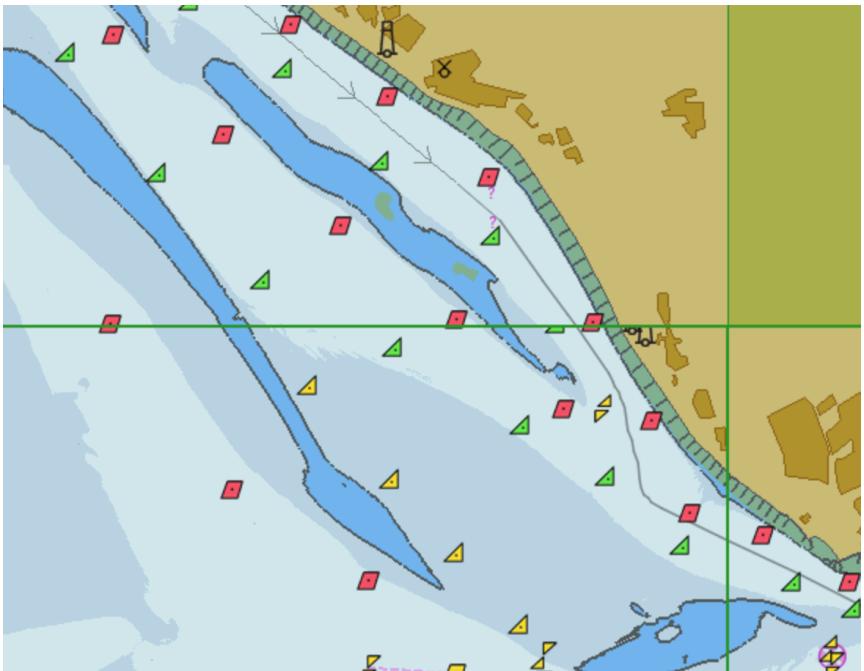
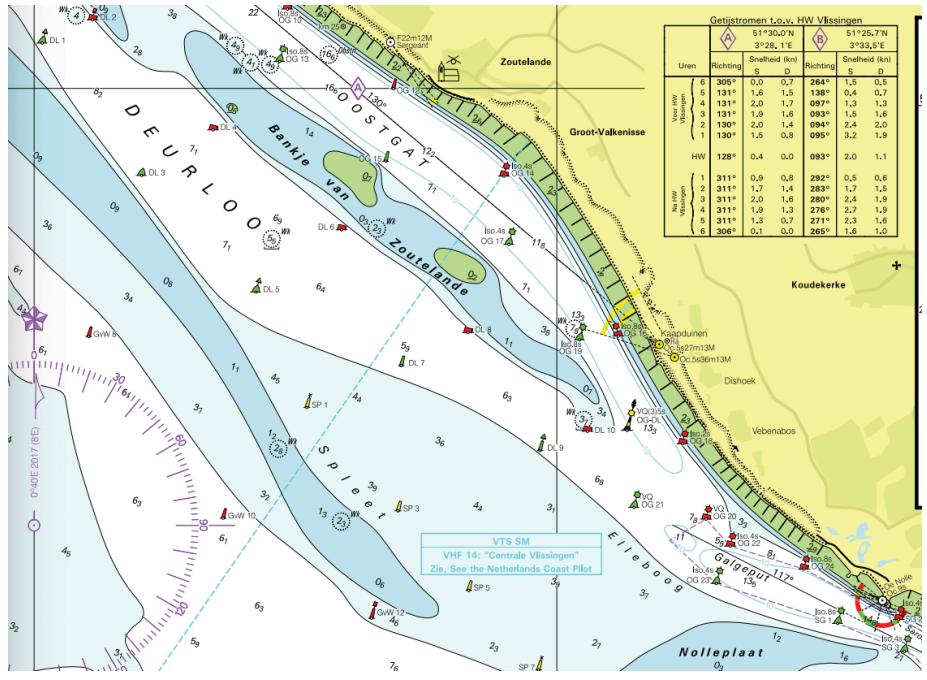






Research objectives

- Integrate *all* constraints in the process
- In such way, all constraints are valid and thus the information is not over-generalised
 - Eliminate human interference
 - Quantify generalisation constraints
 - Evaluate metrics directly within the process
 - Apply operators locally, rather than globally



Research questions

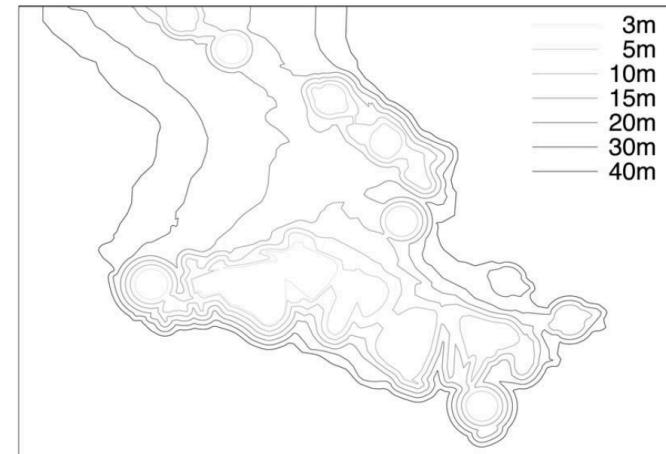
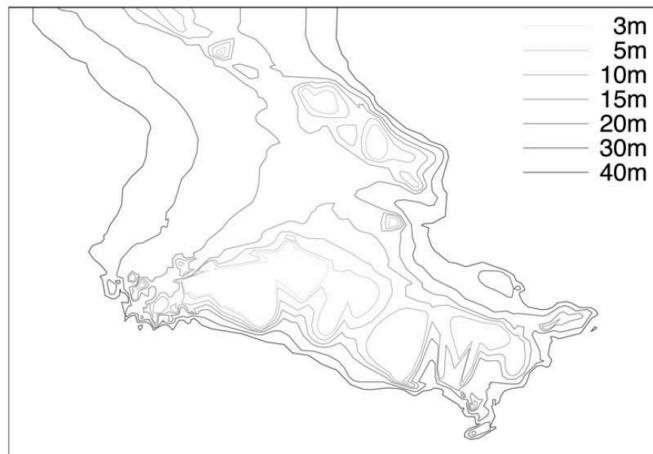
- To what extent can we locally steer generalisation operators to account for cartographic constraints, in a surface-based isobath generalisation method?
 - What are the **minimum legibility constraints** for navigational isobaths, cartographic and legally at different chart scales?
 - How can we quantify the cartographic constraints into **local surface metrics**?
 - What is the effect of applying different **local operators** on the global surface, and how can this be exploited?
 - What are valid and realistic assumptions on **input data** in the field of application?
 - How can the extracted features be **validated** and does the method perform better than available alternatives?

Related work

- Line-based generalisation
- Surface-based generalisation
 - Navigational surface
 - Voronoi-based approach
- Region graph

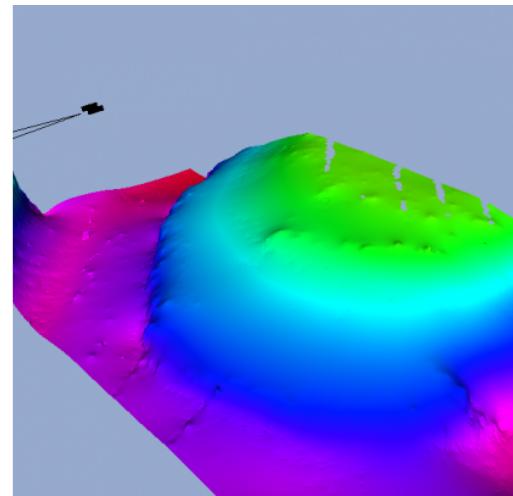
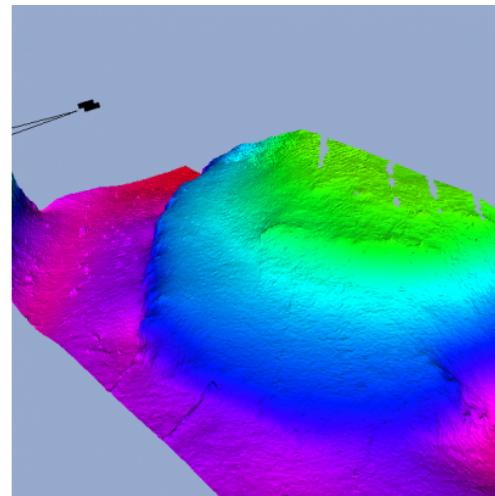
Line-based generalisation

- Multi-agent system
 - Rules and operators
- Lack a connection with survey data



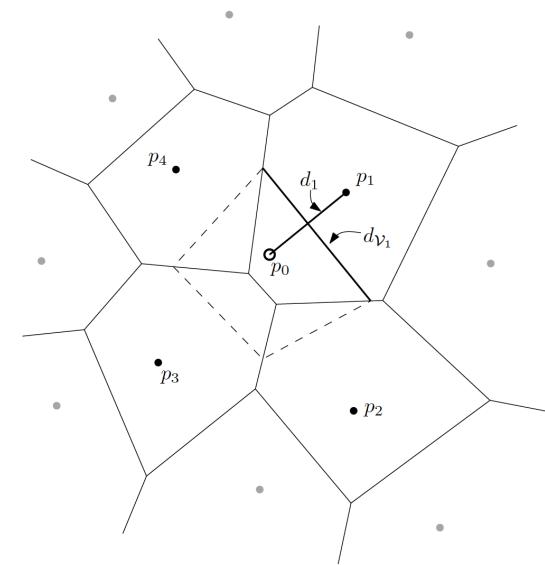
Navigational surface

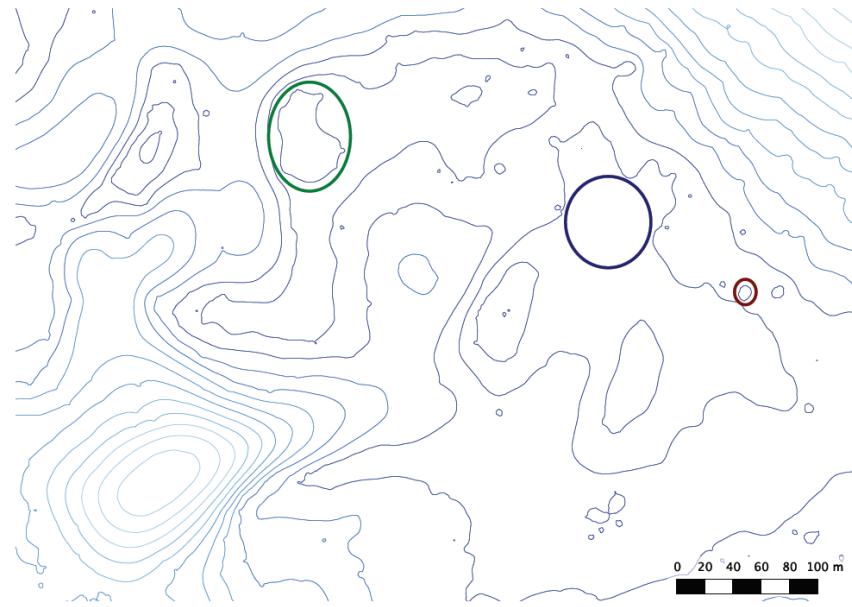
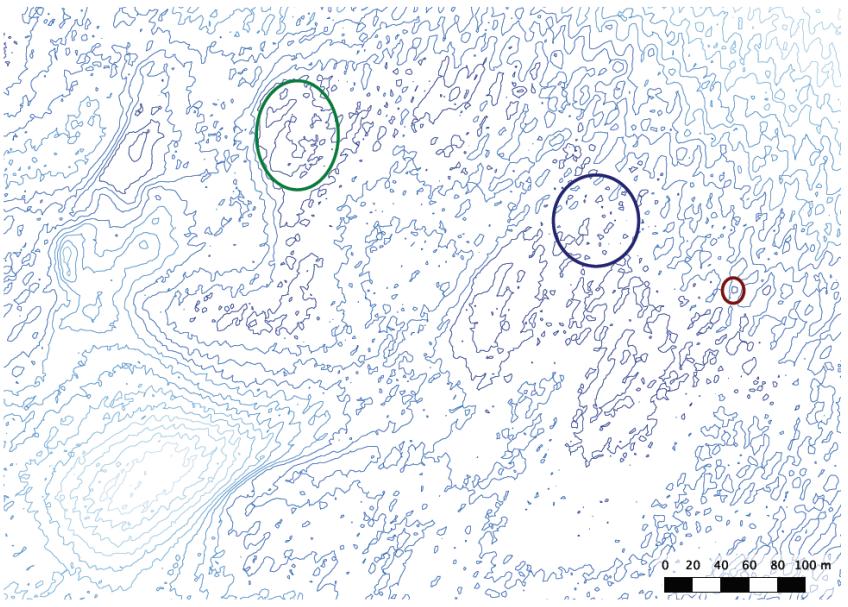
- Raster or vector (triangulation)
- Survey data can be integrated
- Moving the surface only upwards guarantees safety



Voronoi-based generalisation

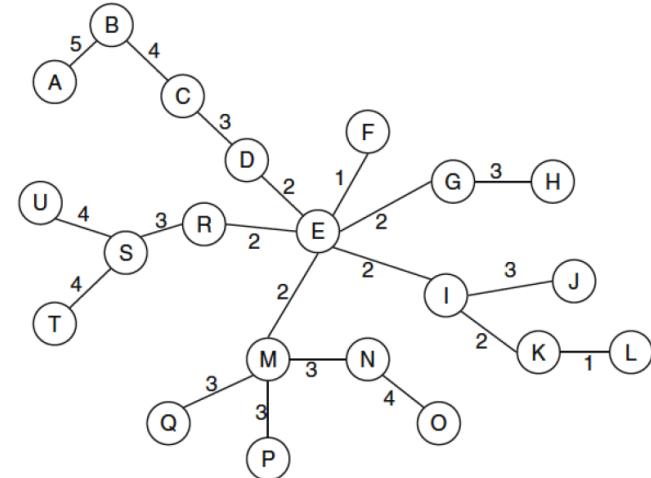
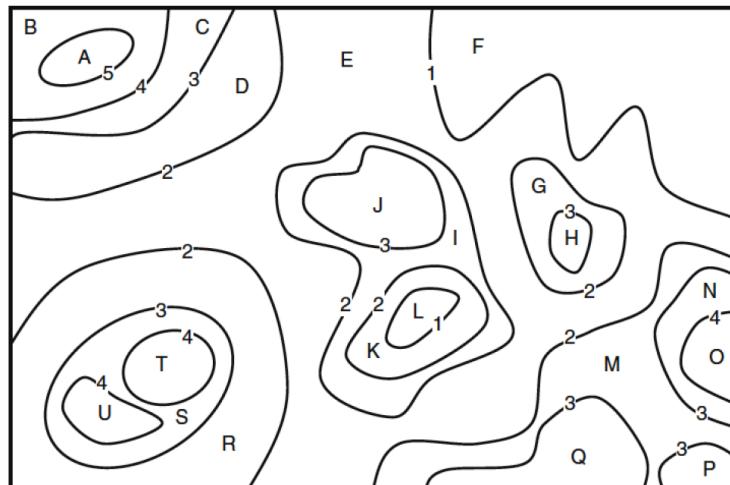
- Generates a smooth surface
 - Thus smooth isobaths
- Laplace interpolation
 - Smooth, local, anisotropic, Parameter-independent, linked to surface
- Iterative approach
 - Smoothing, densification
- When is it good enough?





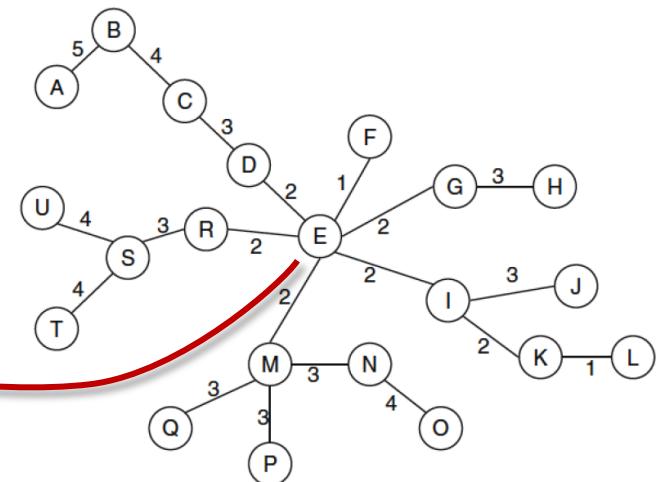
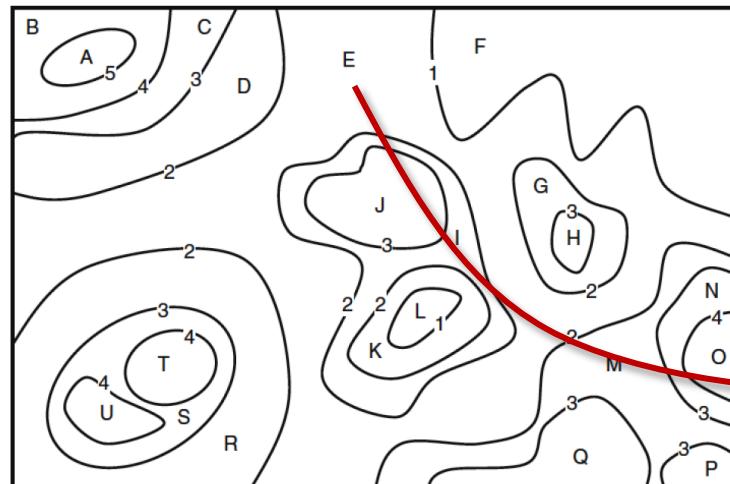
Region graph

- Establish relations between isobaths (not the triangulation)
- Region graph based on inter-isobath area
- Isobaths implicitly defined: edges



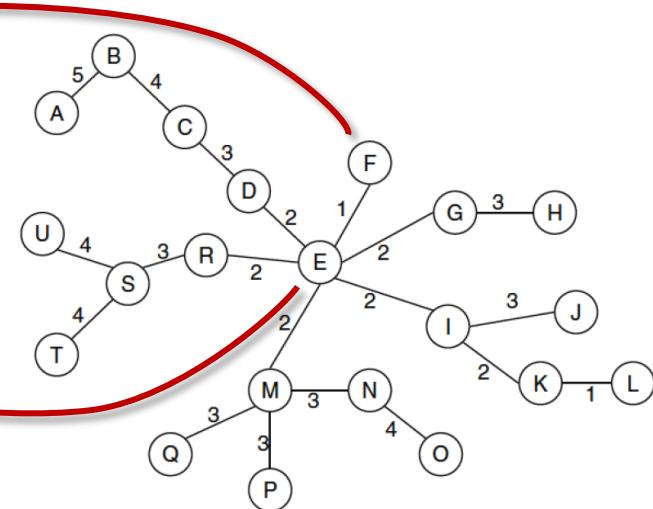
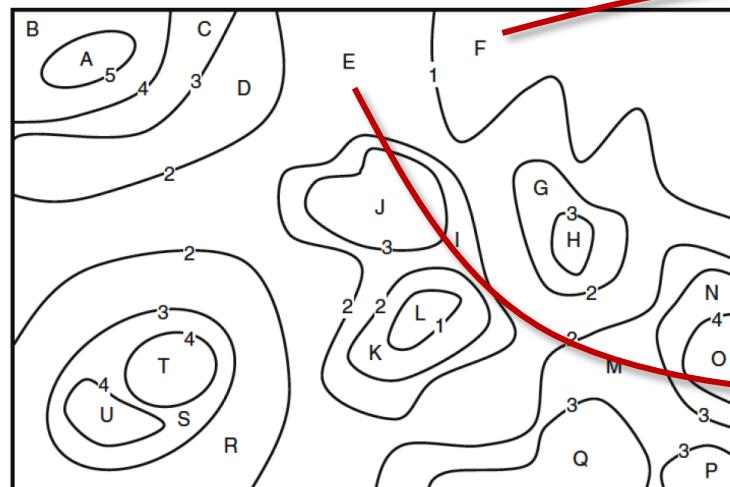
Region graph

- Establish relations between isobaths (not the triangulation)
- Region graph based on inter-isobath area
- Isobaths implicitly defined: edges



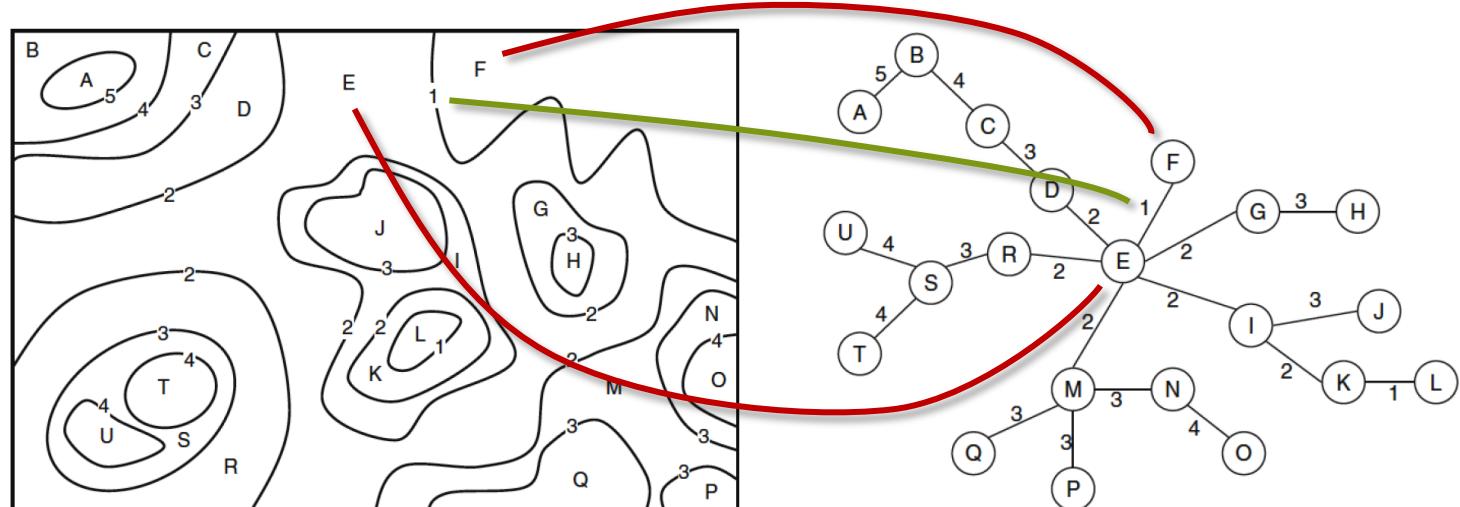
Region graph

- Establish relations between isobaths (not the triangulation)
- Region graph based on inter-isobath area
- Isobaths implicitly defined: edges



Region graph

- Establish relations between isobaths (not the triangulation)
- Region graph based on inter-isobath area
- Isobaths implicitly defined: edges



Methodology

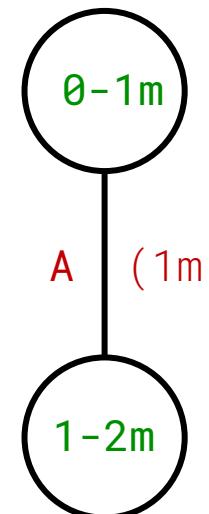
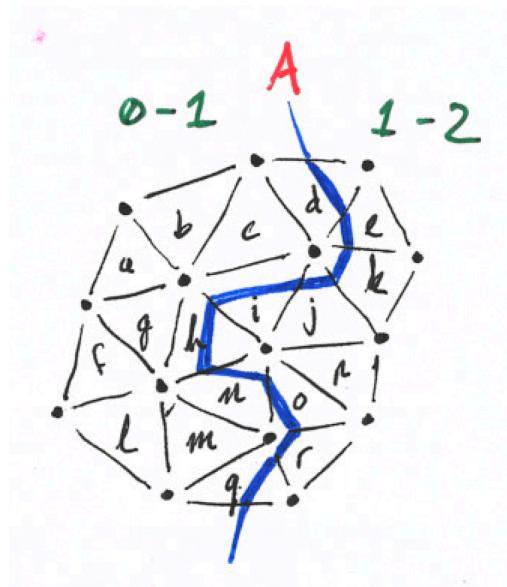
- Link data > triangulation > isobaths
 - Why and where to apply an operator?
1. Establish TIN surface
 2. Define quantified metrics, related to the generalisation constraints
 - a. Find non-valid features, or parts of it
 - b. Select operator to overcome this
 - c. Select region to be applied
 - d. Apply operator on this region
 3. Repeat step 2, until all are valid
 4. Extract information of interest
 5. Validate overall product

Methodology: Objectives

- Pre-defined isobath values
 - *IHO Standard Series*
- Pre-defined compilation scales
 - *NLHO Purposes*
- Limited metrics
 - Smoothness
 - Size
 - Bends
 - Passages

Methodology: Data structure

- Triangle region graph
 - Links the data, triangulation and isobaths together



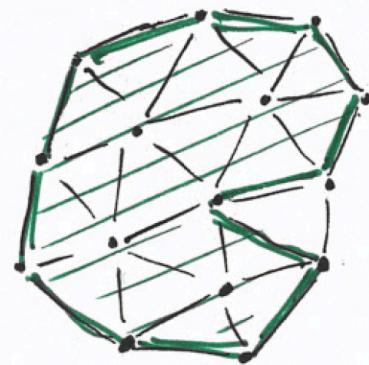
[a, b, c, d, e, f, g, h, i, j, k, l, m, n, o, q, r]

[d, e, h, i, j, k, n, o, q, r]

[d, e, h, i, j, k, n, o, p, q, r]

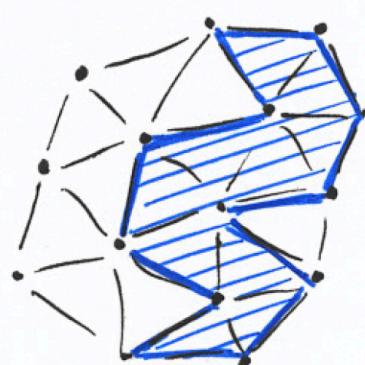
Intersection

Region 1
(node)



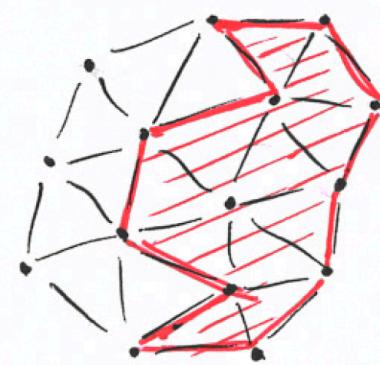
A

Isobath 1-2
(edge)



B

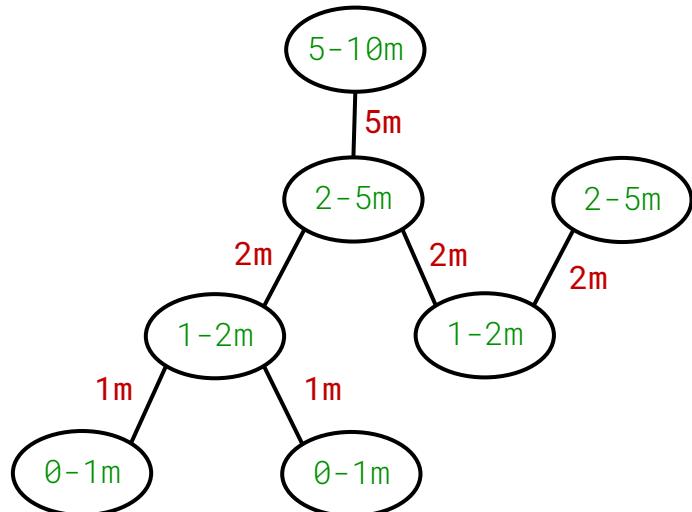
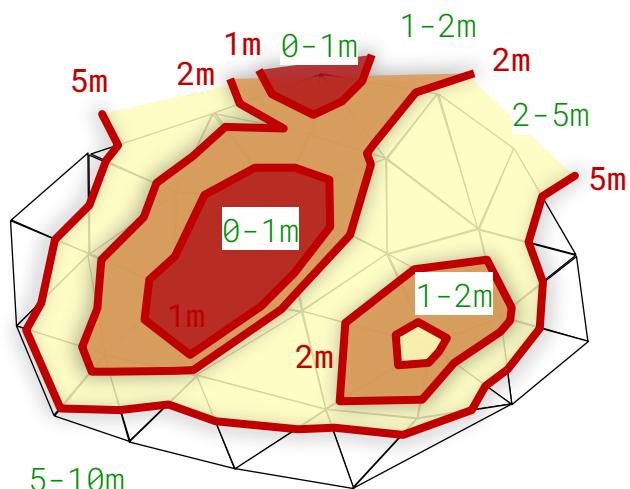
Region 2
(node)



C

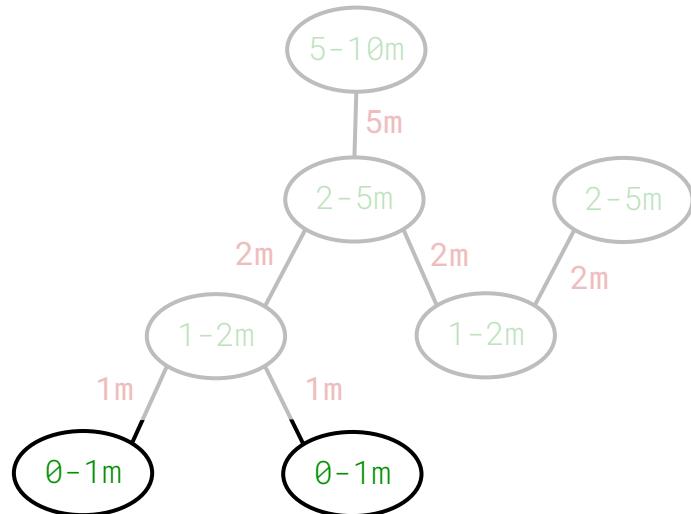
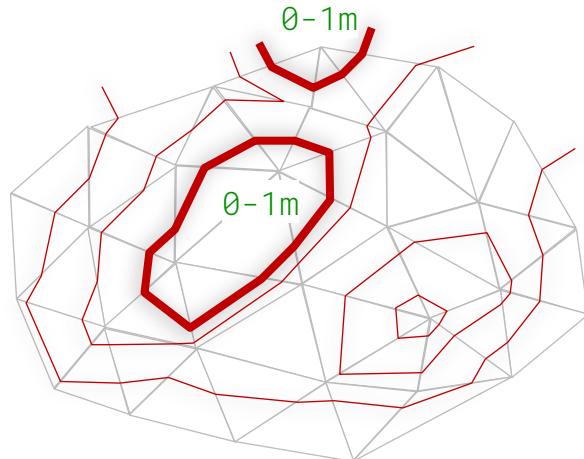
Methodology: Process

1. Construct triangle region graph



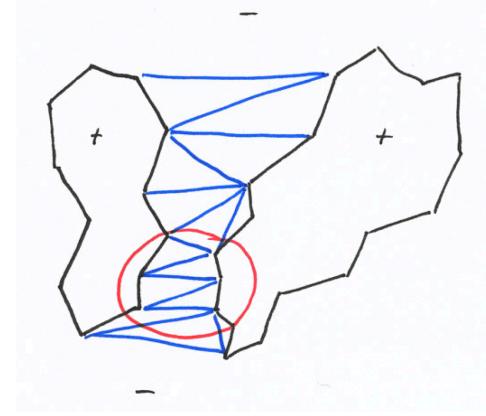
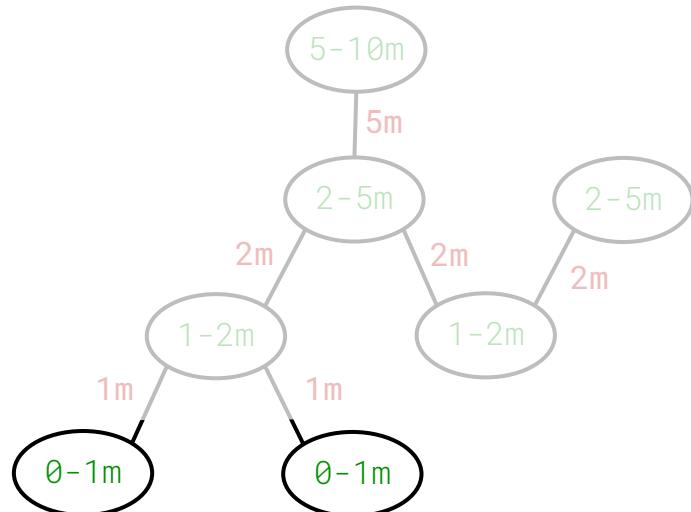
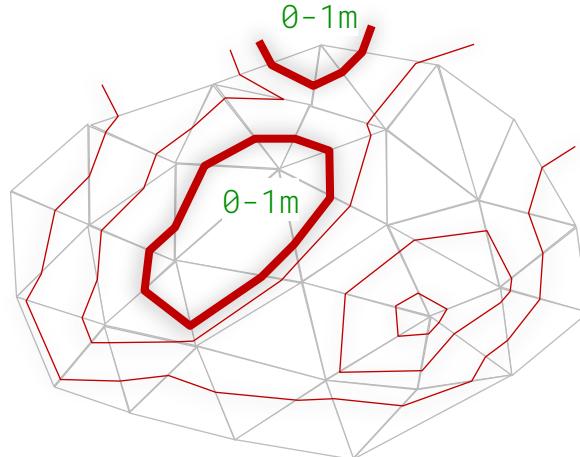
Methodology: Process

2. Start at shallowest node



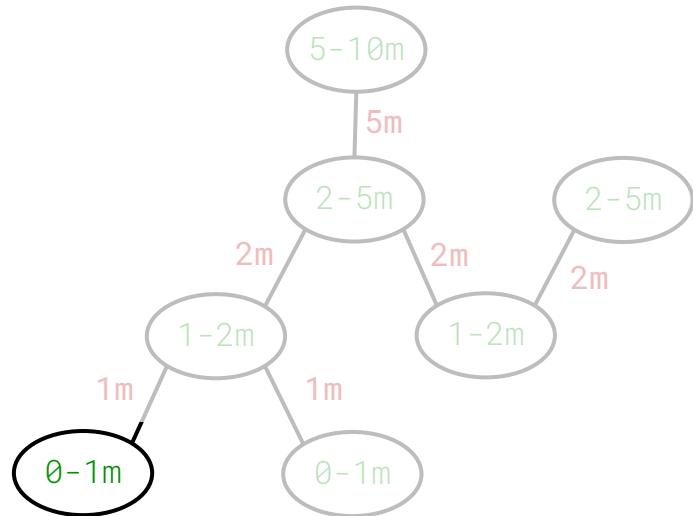
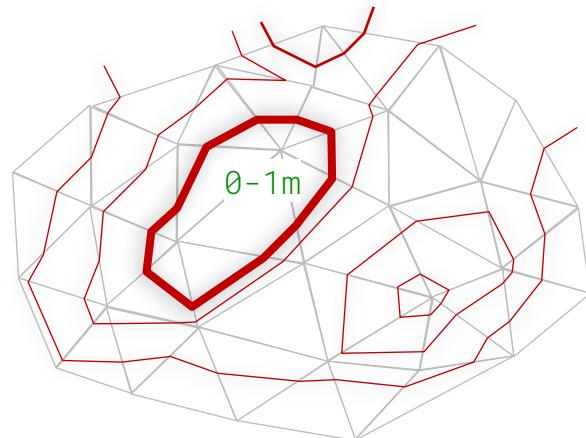
Methodology: Process

2. Start at shallowest node
 - a. Two same-level nodes?
 - > check separation
 - > if critical: no further generalisation



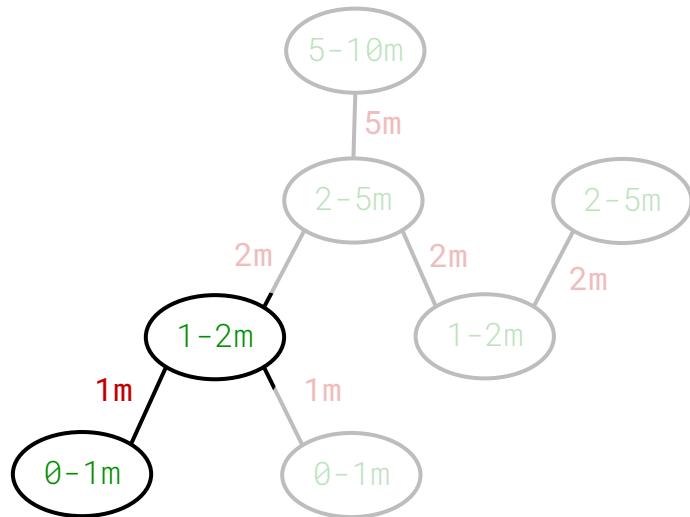
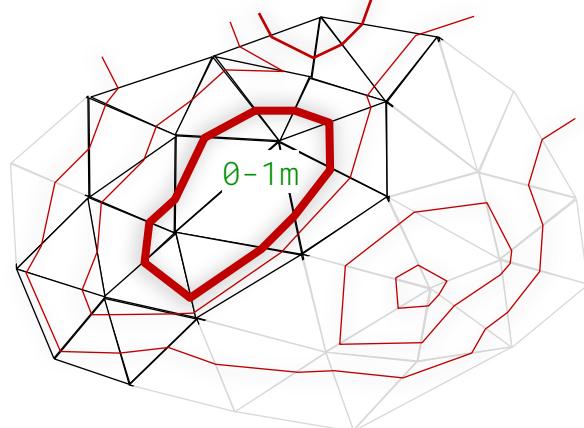
Methodology: Process

2. Start at shallowest node
 - b. Check area of the node



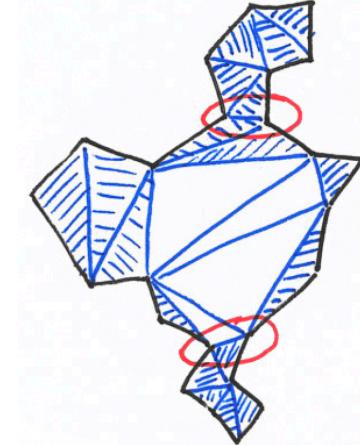
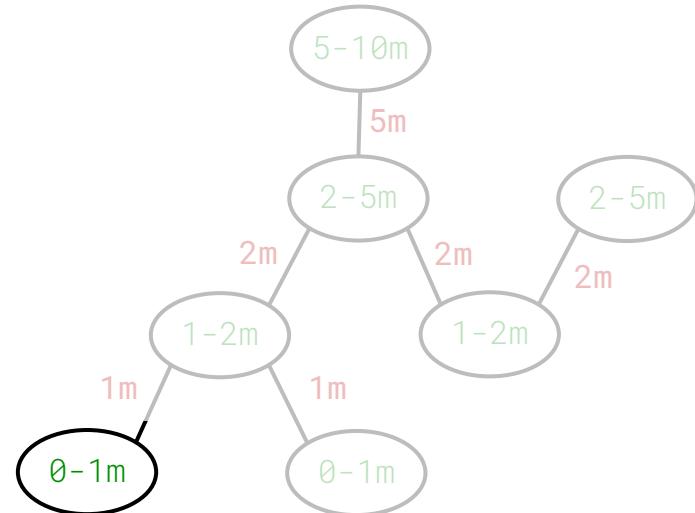
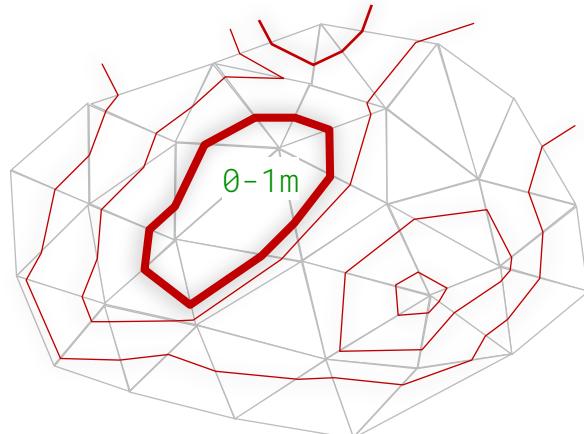
Methodology: Process

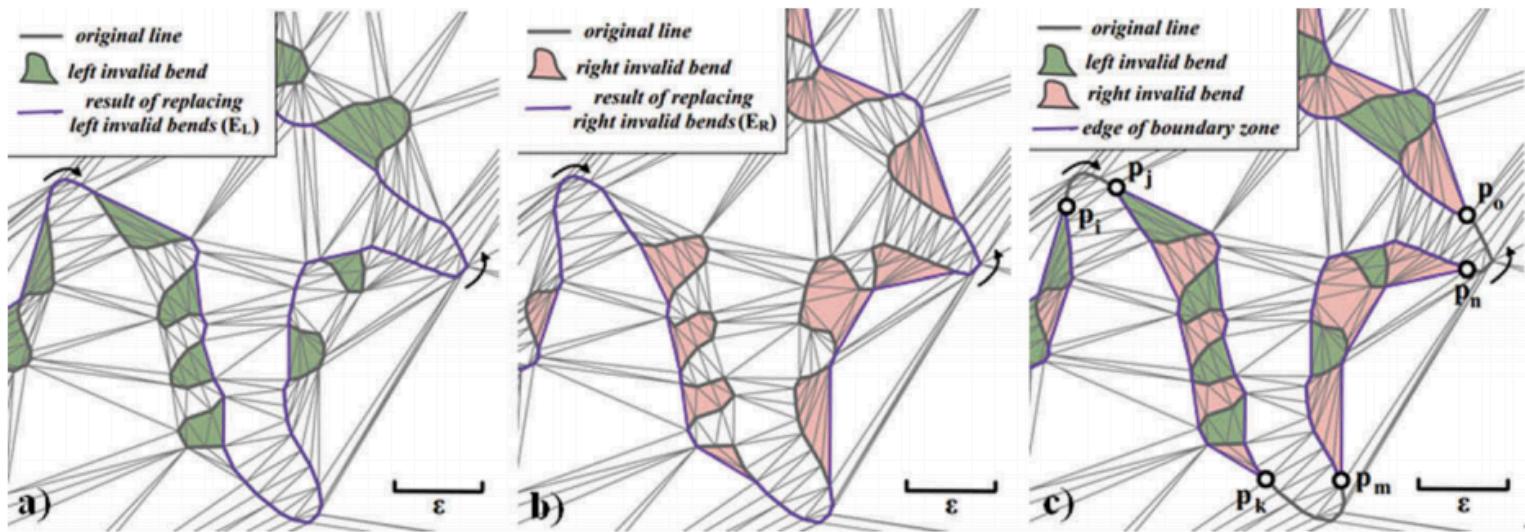
2. Start at shallowest node
 - b. Check area of the node
if too small:
> smoothen region_self + region_deeper



Methodology: Process

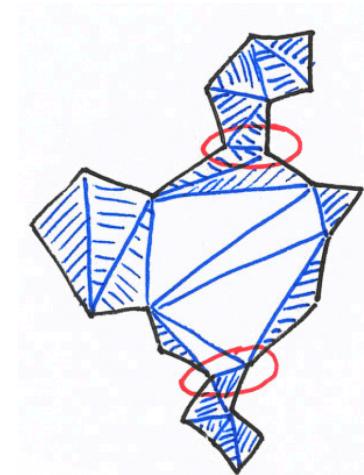
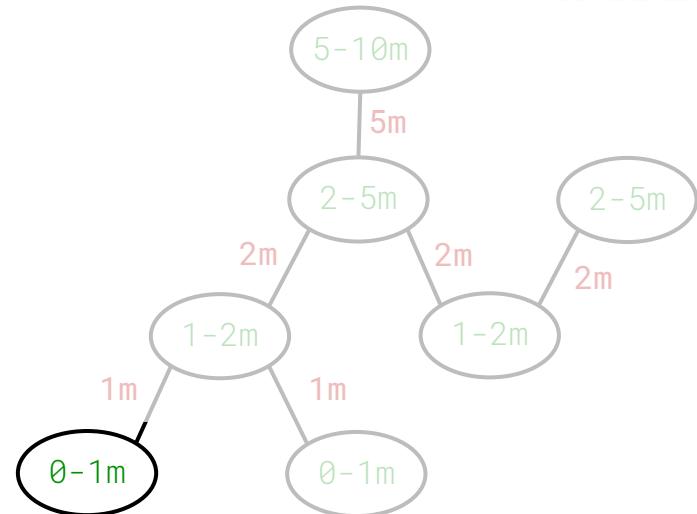
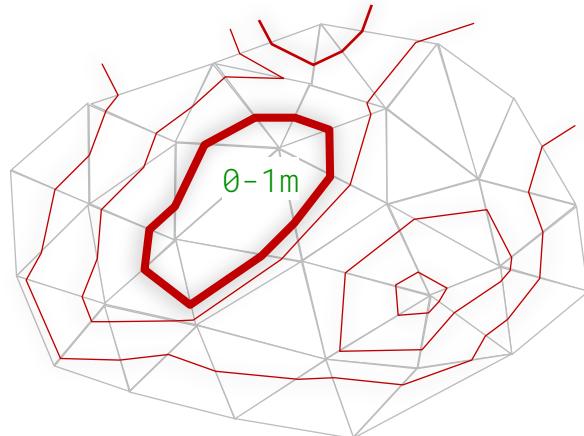
2. Start at shallowest node
 - c. Check outward bends





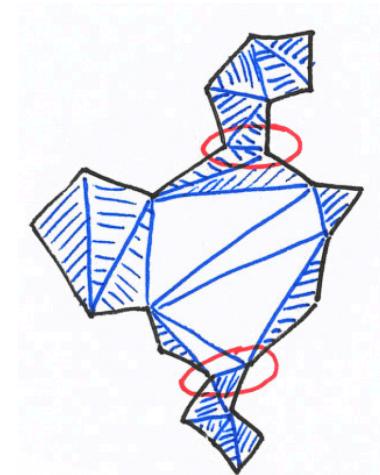
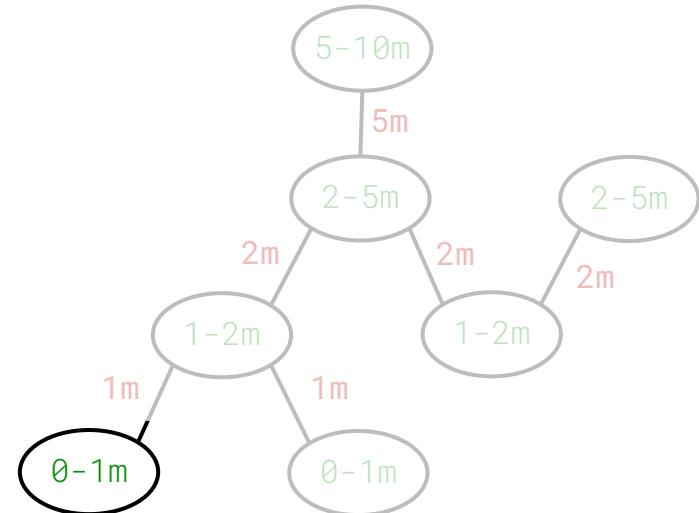
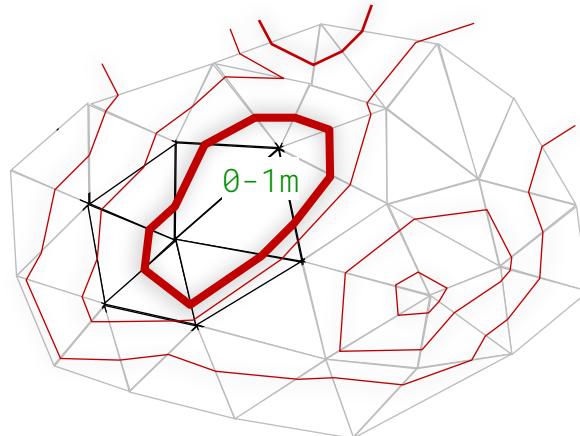
Methodology: Process

2. Start at shallowest node
 - c. Check outward bends



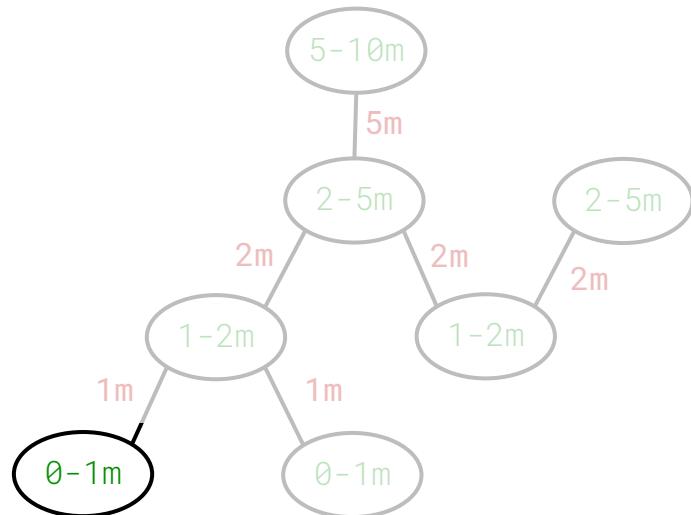
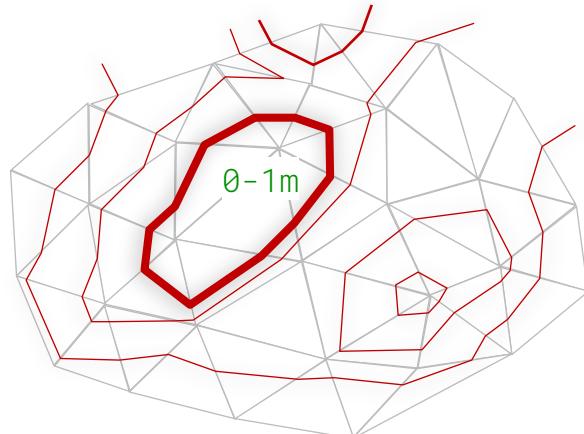
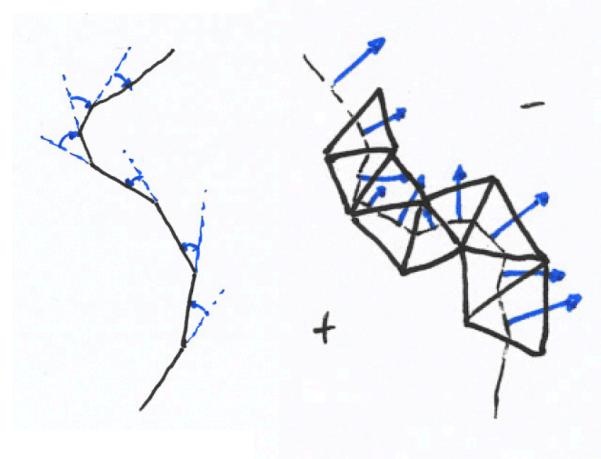
Methodology: Process

2. Start at shallowest node
 - c. Check outward bends if invalid:
> smoothen local region



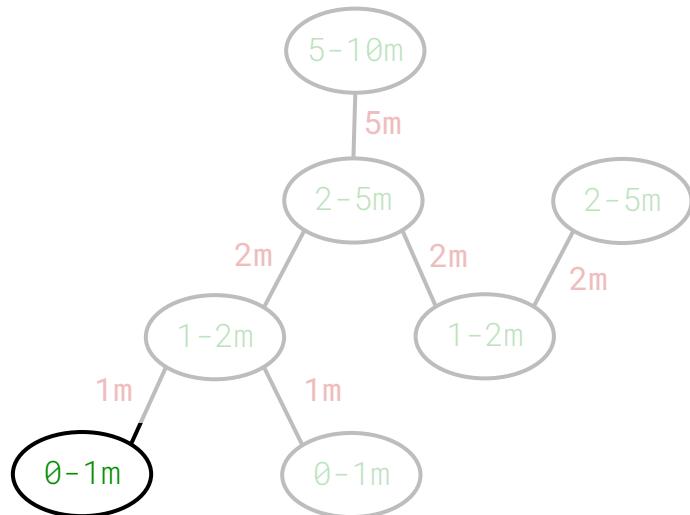
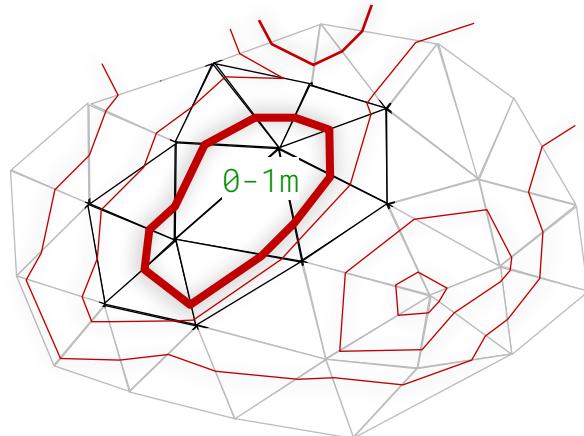
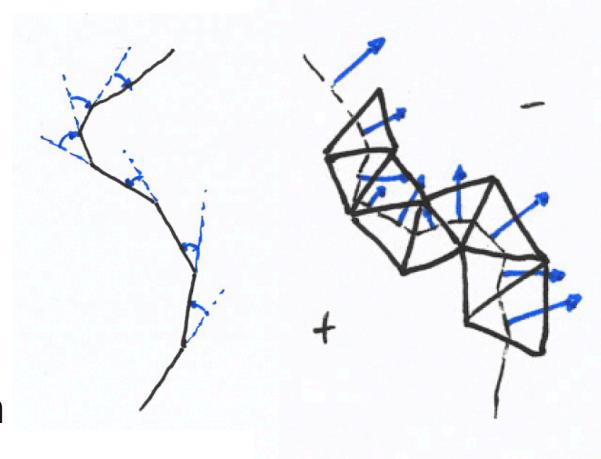
Methodology: Process

2. Start at shallowest node
 - d. Check overall smoothness



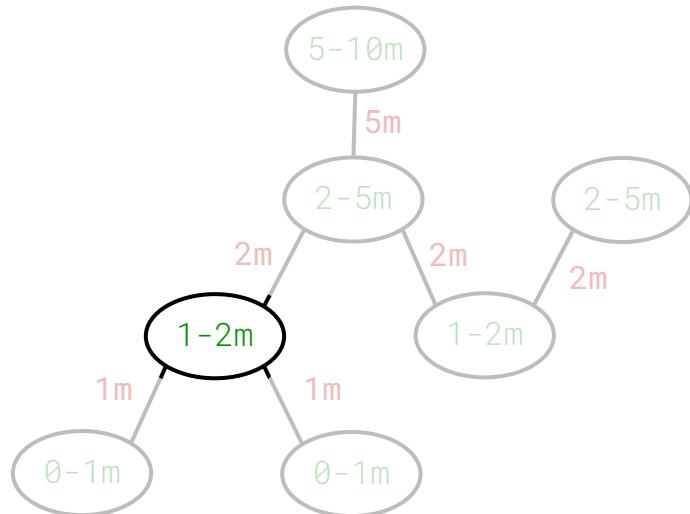
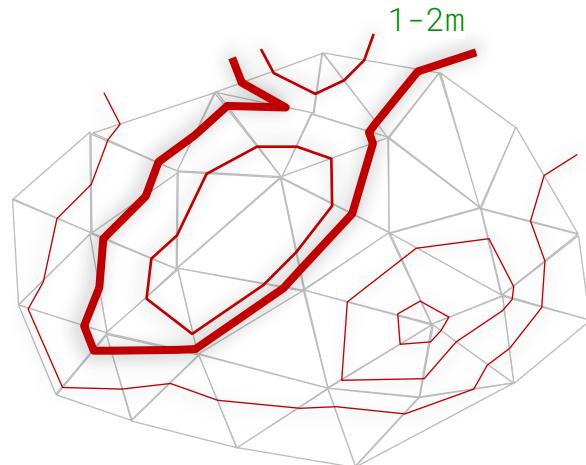
Methodology: Process

2. Start at shallowest node
 - d. Check overall smoothness
if invalid:
> smoothen or densify region_isobath



Methodology: Process

1. Construct triangle region graph
2. Start at shallowest node
3. Traverse to (next/) deeper node

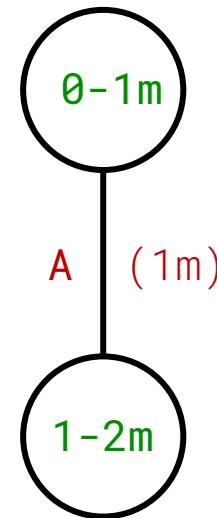
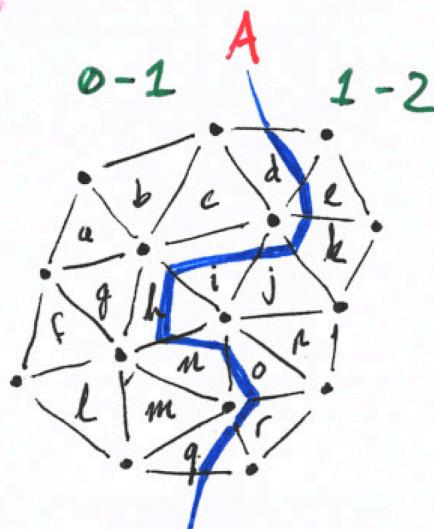


Methodology: Process

1. Construct triangle region graph
2. Start at shallowest node
 - a. Check same-level separation
 - b. Check area
 - c. Check bends
 - d. Check smoothness
3. Traverse to (next/) deeper node
4. Repeat until every node is visited and valid
5. Extract information of interest

Methodology: Output

- Isobaths
 - Triangles intersecting an isobath already indexed



triangles

[a, b, c, d, e, f, g, h, i,
j, k, l, m, n, o, q, r]

[d, e, h, i, j, k, n, o,
q, r]

[d, e, h, i, j, k, n, o,
p, q, r]

Methodology: Validation

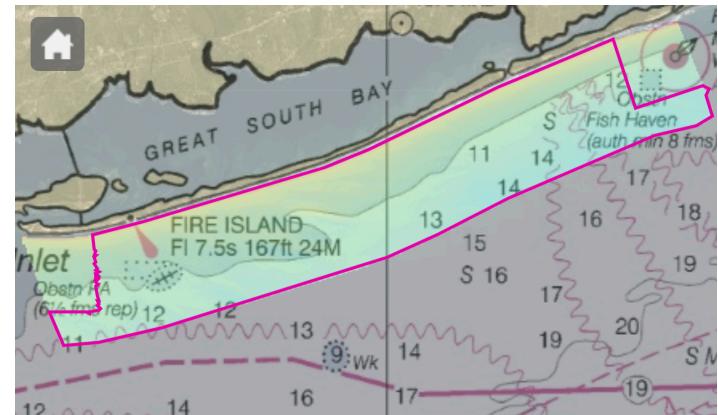
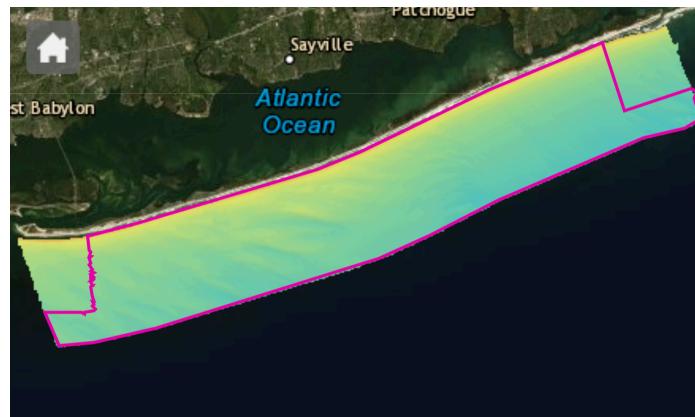
- Safety and topology: hard constraints
- Morphology: statistical measures on vertical displacement
- Legibility: smoothness, and overall readability
 - Maybe feedback from cartographers
- Relative performance
 - Morphology and legibility
 - w.r.t. existing charts
 - w.r.t. current implementation

Tools and data

- Python 3 programming
- QGIS visualisation
- GDAL converting ENCs
- Triangulation library
 - Topological structure
 - Vertex updating
 - Pointers from the region graph to triangles
 - Constrained Delaunay
 - Voronoi diagram

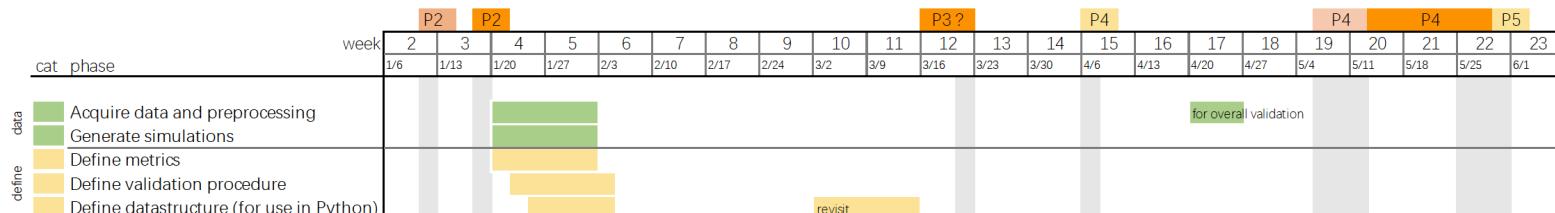
Tools and data

- Survey data
- Simulated survey data
- Isobaths (reference/compare)



Planning

- Iterative and modular approach
- First define metrics and other theory
- Minimal working prototype
 - Slowly extending it
- P3:
 - Metric calculation, region selection, operator strategy



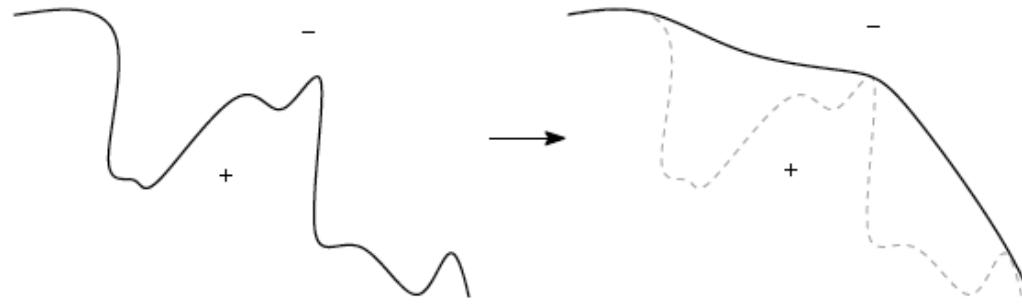
Thank you!

Willem van Opstal

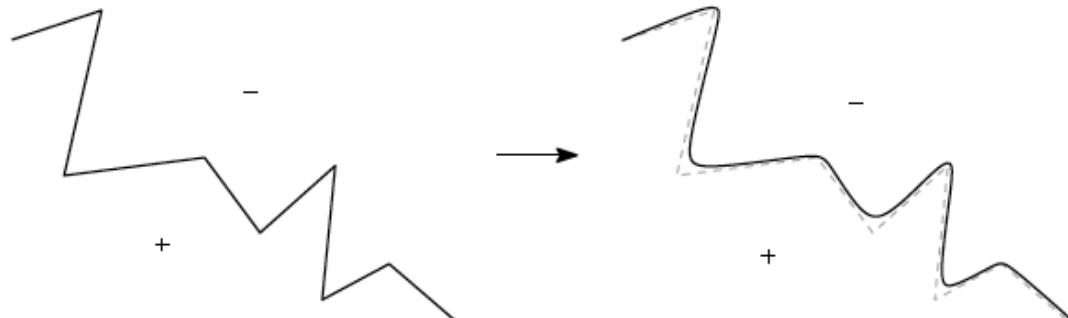
Martijn Meijers (1st mentor)

Ravi Peters (2nd mentor)

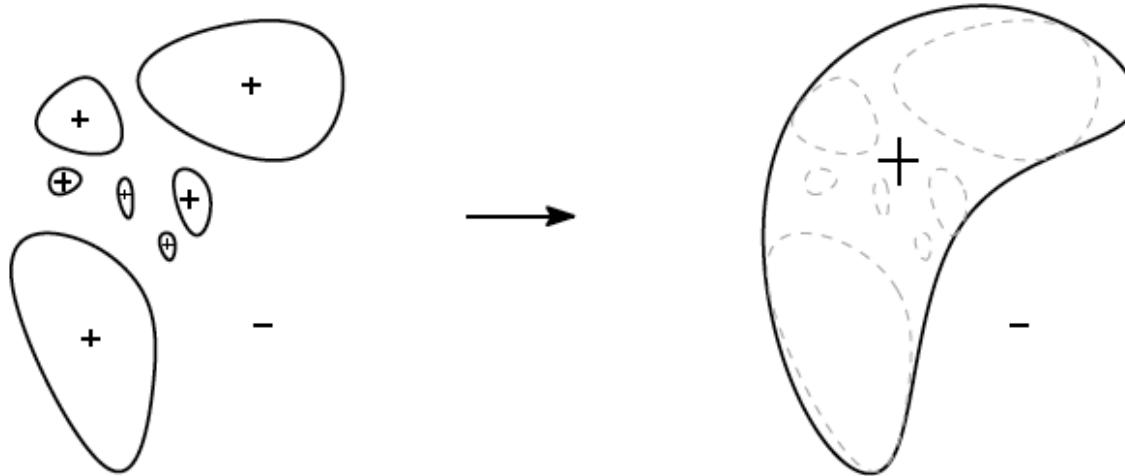
January 16th 2020



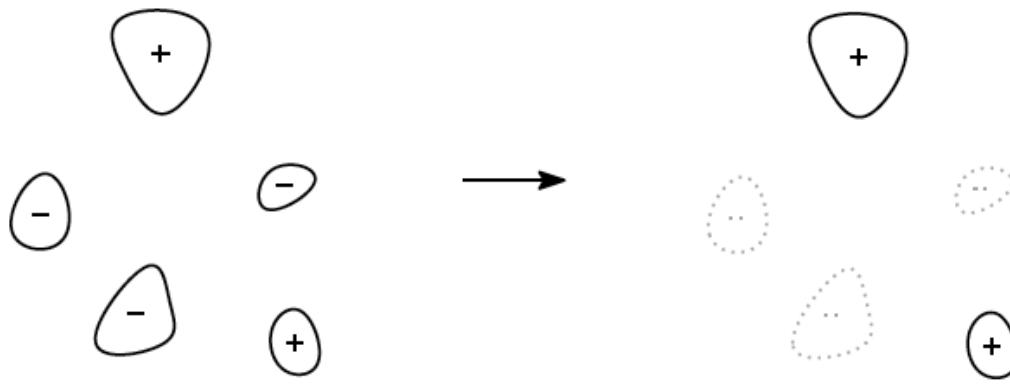
(a) Simplification



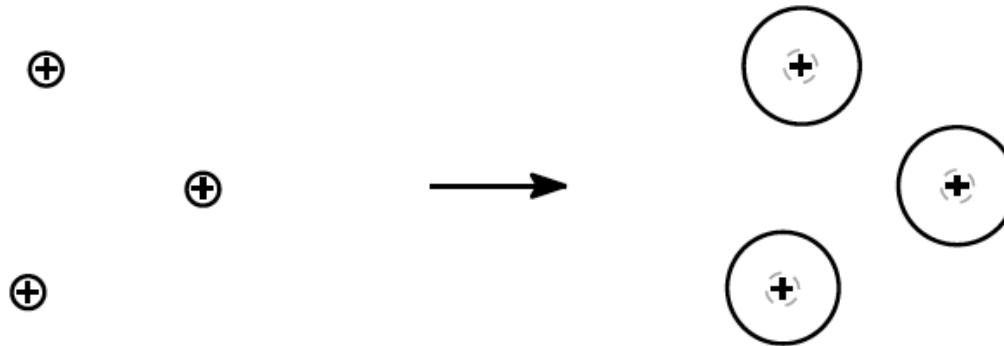
(b) Smoothing



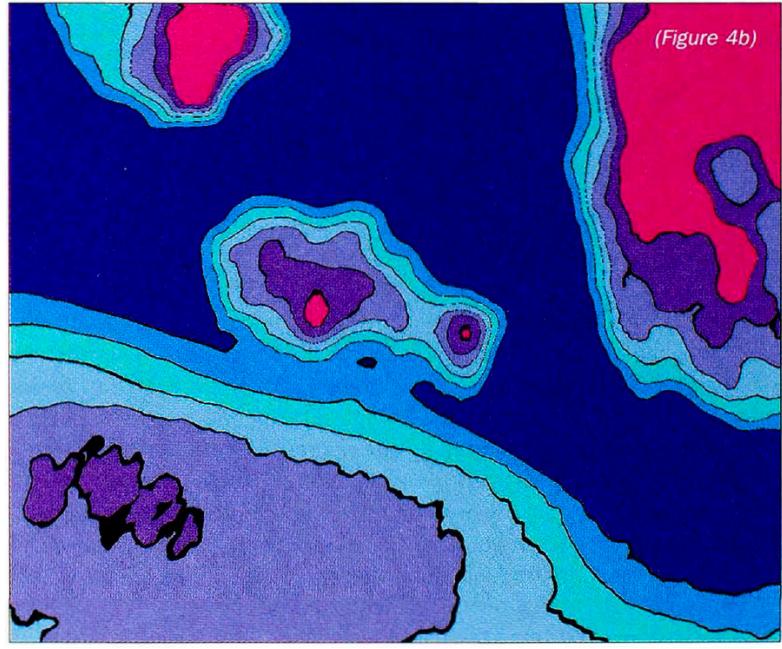
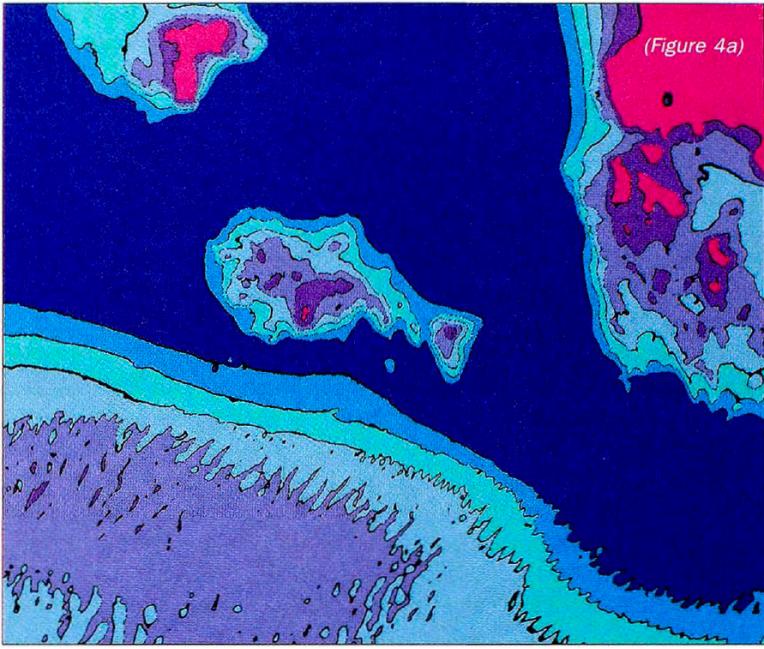
(c) Aggregation



(d) Omission



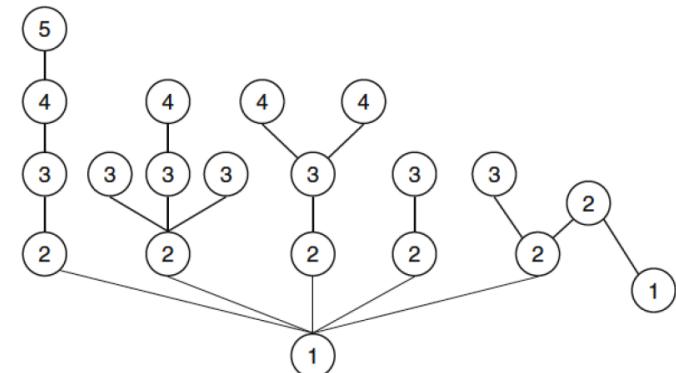
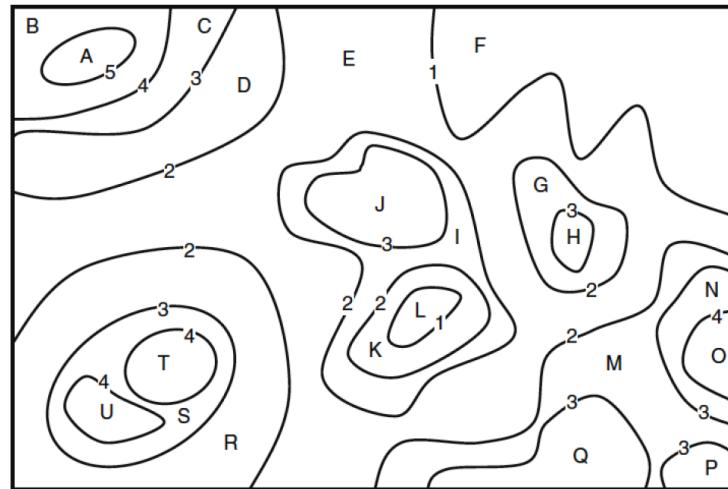
(e) Enlargement



Contour tree

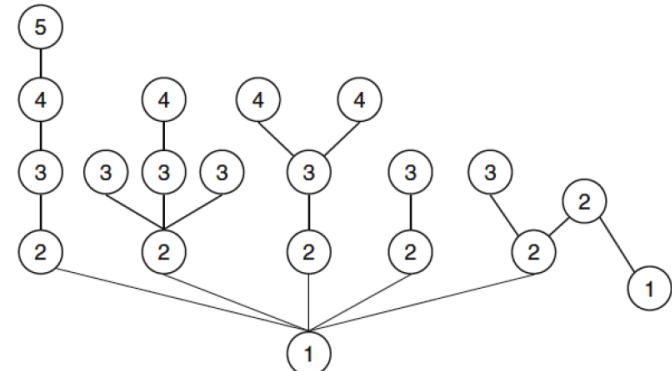
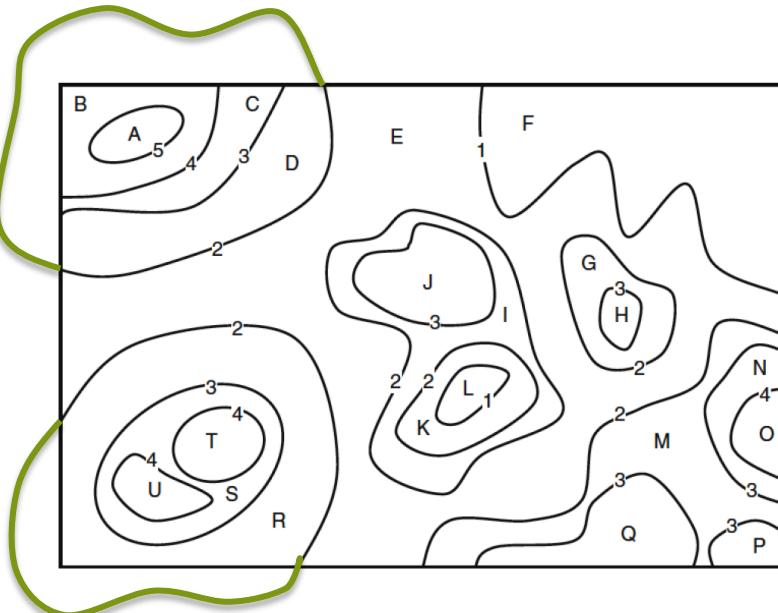
- Establish relations between isobaths

(not the triangulation)



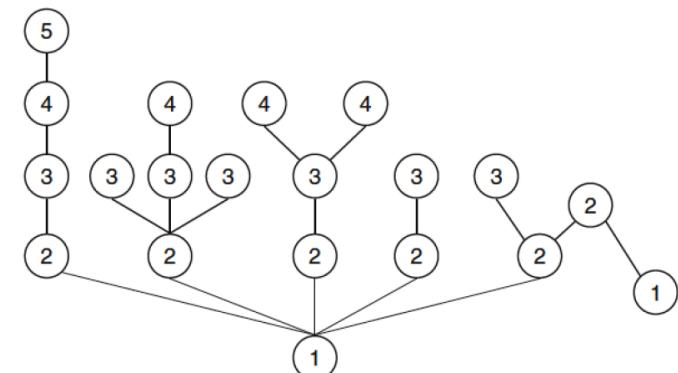
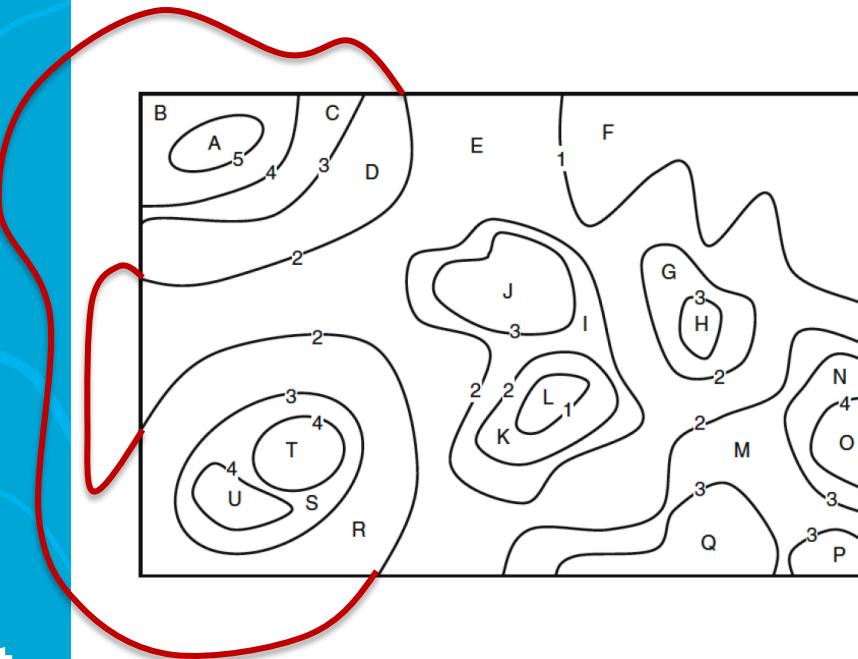
Contour tree

- Contour tree cannot handle boundaries, difficult to create



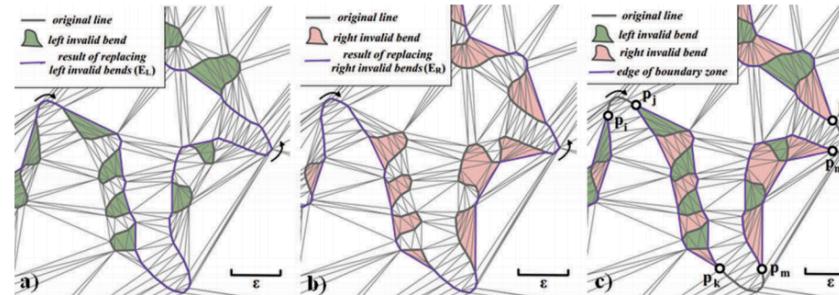
Contour tree

- Contour tree cannot handle boundaries, difficult to create



Shape metrics

- Surface metrics
 - Triangle area, orientation, aspect ratio
- Shape metrics
 - Overall shape of a feature
- Line metrics
 - Smoothness
- Bend zone detection
 -

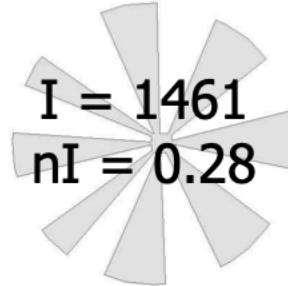


index

$I = 7417$
 $nI = 0.87$

normalized index

$I = 7344$
 $nI = 1.00$



$I = 1700$
 $nI = 0.28$

$I = 1200$
 $nI = 0.29$

