Learning topological operations on meshes with application to block decomposition of polygons

Per-Olof Persson, Arjun Narayanan, Lewis Pan

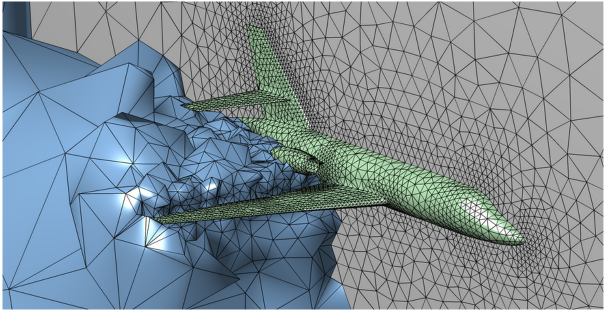
Department of Mathematics, University of California, Berkeley
Mathematics Department, Lawrence Berkeley National Laboratory

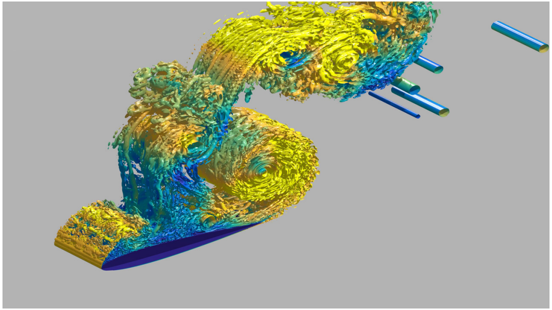
Tetrahedron VII: Seventh Workshop on Grid Generation for Numerical Computations

October 11, 2023

Motivation: High-Order Methods on Unstructured Meshes

- Widely believed that *high-order accurate methods* will be required for challenging simulations (turbulent flows, wave propagation, etc)
- In addition, *fully unstructured meshes* are necessary to handle complex geometries, with adapted resolution and full automation
- Goal: Develop *robust*, *efficient*, and *accurate* high-order methods





A Face Upwinded Spectral Element Method (FUSE) for Conservation Laws

Scientific Achievement

A new stabilization scheme for high-order continuous Spectral Element Methods which is provable convergent up to any order.

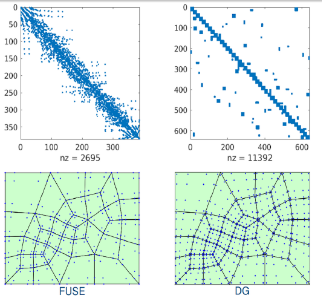
Significance and Impact

The work has the potential to drastically improve the performance of high-order methods, which are widely believed to be required for accurate predictions of turbulent flows and problems with waves and non-linear interactions.

Technical Approach

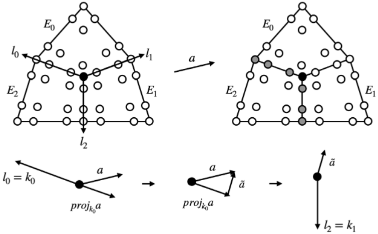
- Most stabilized schemes for fluids and other conservation laws are based on discontinuous formulations (e.g., the discontinuous Galerkin method)
- A remarkably simple way to stabilize continuous methods: Inspired by finite difference methods, choose the full upwind stencil only for face nodes
- Provably high-order convergent for a non-standard node distribution
- In addition, a line-based sparsity patterns bring the Jacobian cost from $\mathcal{O}(p^D)$ to $\mathcal{O}(pD)$, for polynomial degree p in D dimensions

PI(s)/Facility Lead(s): Per-Olof Persson, LBNL Math Group
ASCR Program: Base Math
ASCR PM: Steven Lee
Publication(s) for this work:
Y. Pan, P.-O. Persson, “A Face-Upwinded Spectral Element Method on Unstructured Quadrilateral Meshes,” *Journal of Computational Physics* (in review)
Y. Pan, P.-O. Persson, “A Stabilized Face-Upwinded High-Order Method for Incompressible Flows,” *Proc. of 2023 AIAA AVIATION,* June 2023.



Sparsity pattern for FUSE vs DG, at polynomial degree 3. Due to continuous fields and line-based sparsity patterns, the Jacobian matrices are more than 4 times cheaper. This effect increases in 3D and for higher degrees.

Face-Upwinding for unstructured grids. For general non-linear roblems, the upwinding directions depend on the solution. Also note the line-based sparsity pattern.

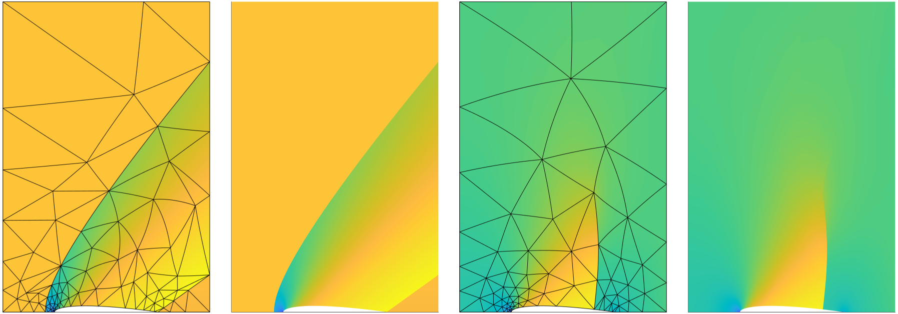


Previous work: HOIST – High-Order Implicit Shock Tracking

- Use full-space optimization to align high-order curved meshes with discontinuities
- Error estimator based on a p-adaptive residual:

$$r^K_{h,p'}(U_{h,p}) := \int_{\partial K} \psi^+_{h,p'} \cdot \mathcal{H}(U^+_{h,p}, U^-_{h,p}, n)\, dS - \int_K F(U_{h,p}) : \nabla \psi_{h,p'}\, dV$$

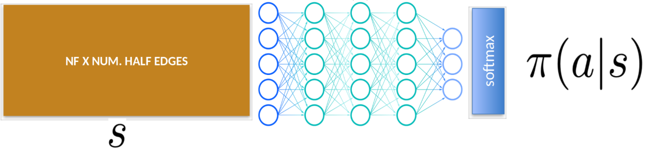
- Minimize $f_{\text{err}}(\boldsymbol{u}, \boldsymbol{x}) := \frac{1}{2}\boldsymbol{R}(\boldsymbol{u}, \boldsymbol{x})^T \boldsymbol{R}(\boldsymbol{u}, \boldsymbol{x})$ to align mesh to shocks
- Solve with efficient SQP solver based on a full-space approach

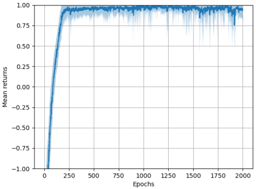


Zahr, Persson, (JCP 2018), Zahr, Shi, Persson (JCP 2020/22), Zahr *et al* (multiple).

Deep Reinforcement Learning for Block Meshing

- Define a "game" for automatic block mesh improvement:
  - "Moves": Local or global topological operations (e.g. "flips")
  - "Score": Measure of irregularity of the mesh $s = \sum_i |\Delta_i|$
- Use a half-edge mesh structure to define a CNN-type network which extends to fully unstructured quadrilateral meshes
- Train on random geometries, using the PPO algorithm on GPUs
- Consistently produces close-to-optimal meshes

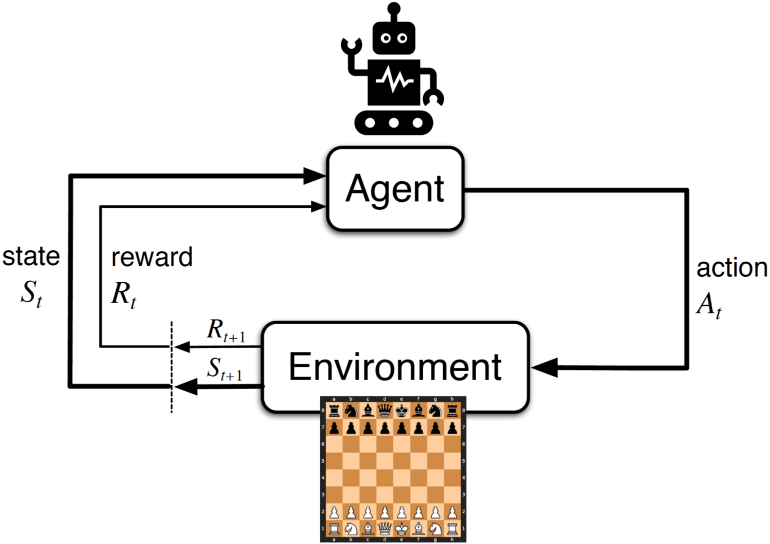




[1] Narayanan, Pan, Persson. *Learning topological operations on meshes with application to block decomposition of polygons*. In review & arXiv:2309.06484.

Live Mesh Demo

Basic idea of reinforcement learning



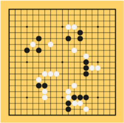
Reinforcement Learning, Solutions Methods

Finite state-space ✓

Finite action-space ✓

Tabular methods
Iterative methods with provable convergence

10^{45} states



10^{170} states

Sampling based methods
- Monte Carlo Tree Search
- Deep RL

Some Useful Terminology

$$\Pi(a_t|s_t, \theta)$$

Policy: Probability distribution over actions

$$P(s_{t+1}|s_t, a_t)$$

State transition probability

$$\tau = s_0, a_0, \ldots, s_H, a_H$$

State – action trajectory

$$R(\tau) = \sum_{t=0}^{H} R(s_t, a_t)$$

Cumulative returns of trajectory

Objective function

$$U(\theta) = \mathbb{E}\left[R(\tau); \Pi_\theta\right] \\ = \sum_\tau P(\tau; \theta) R(\tau)$$

$$\theta^* = \arg\max_\theta U(\theta)$$

Estimating gradient of objective

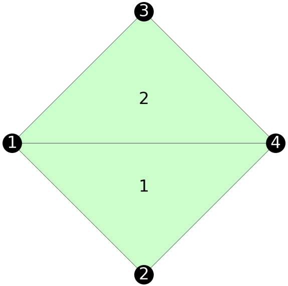
$$U(\theta) = \sum_{\tau} P(\tau;\theta) R(\tau)$$

$$\nabla_\theta U(\theta) = \sum_{\tau} \nabla_\theta P(\tau;\theta) R(\tau)$$

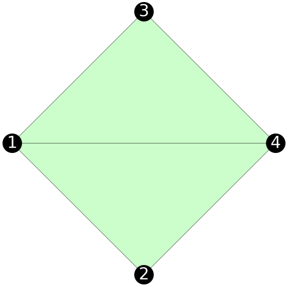
$$= \sum_{\tau} P(\tau;\theta) \frac{\nabla_\theta P(\tau;\theta)}{P(\tau;\theta)} R(\tau)$$

$$= \mathbb{E}\left[\nabla_\theta \log(P(\tau;\theta)) R(\tau)\right] \quad \approx \frac{1}{m} \sum_{i=1}^{m} \boxed{\nabla_\theta \log(P(\tau^{(i)};\theta))} R(\tau^{(i)})$$

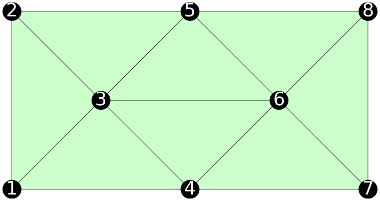
Mesh editing operations - triangles



Edge-flip

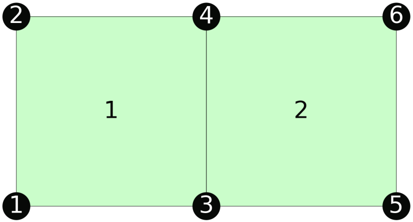


Edge-split

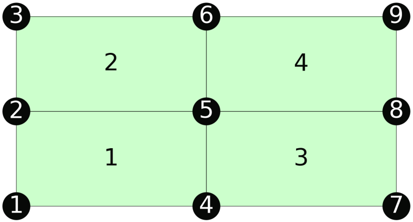


Collapse

Mesh editing operations - quadrilaterals, local

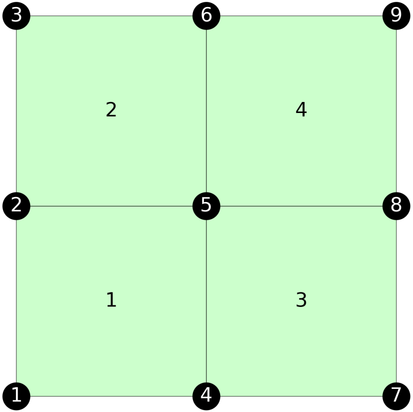


Flip

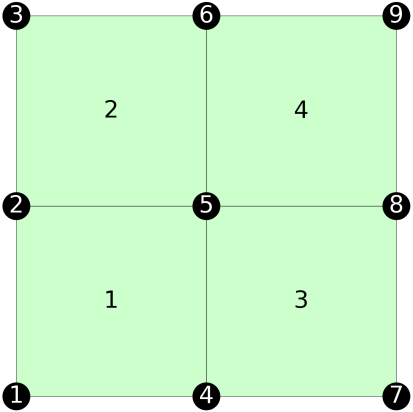


Split-Collapse

Mesh editing operations - quadrilaterals, global



Global Split



Global Cleanup

Objective: minimize vertex irregularity

Given:

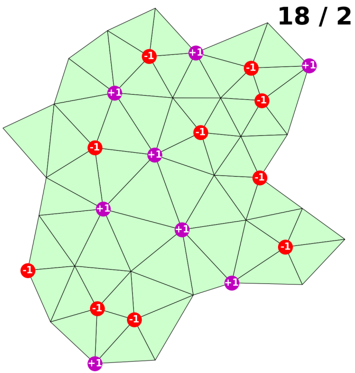
- Mesh m
- Desired degree of vertices d^*:

$$d^* = \begin{cases} 360/\alpha & \text{interior vertex} \\ \max\left(\lfloor \theta/\alpha \rceil + 1, 2\right) & \text{boundary vertex} \end{cases}$$

  where $\alpha = 60$ for triangles, 90 for quads, and θ is the angle of a boundary point.

- Define $\Delta_i = d_i - d_i^*$

minimize $s = \sum_i |\Delta_i|$

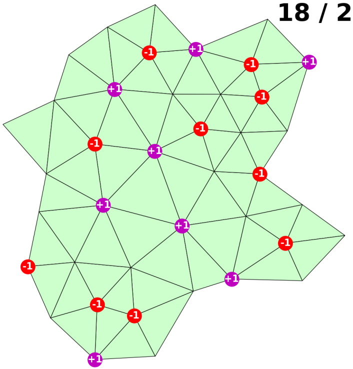


Lower bound on objective function

Note that:

- $s^* = \left|\sum_i \Delta_i\right| \leq \sum_i |\Delta_i| = s$
- s^* is invariant under mesh edits.

This means s^* is a bound on the best possible improved mesh $\implies$ use for a normalized optimality score.



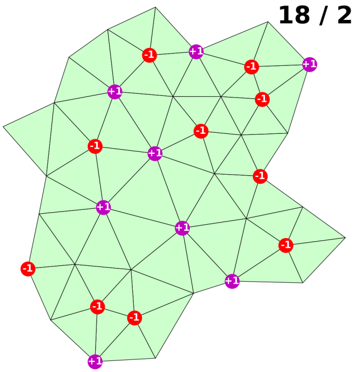
Challenging, unstructured problem

The problem poses several challenges:

- Discrete decisions
- Fully unstructured
- Dynamic data-structure

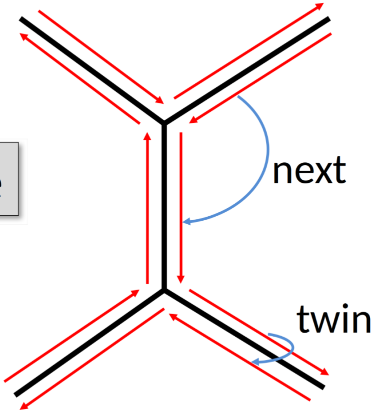
Solution methods need to be able to:

- Represent and understand mesh topology
- Efficiently implement mesh edits



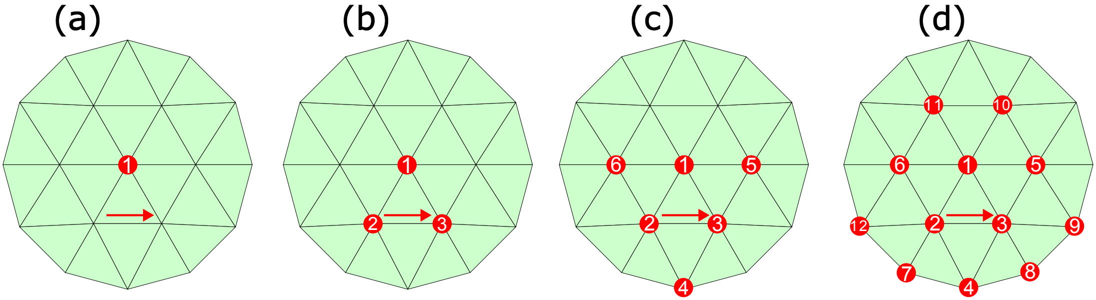
Half-edges represent topology in a structured way

Action: Half-edge + type



Half-edge operations used to represent state

Template: Ordered sequence of vertices around each half-edge



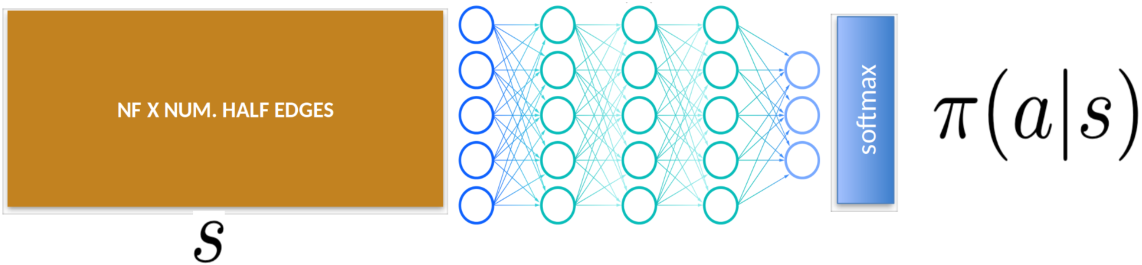
In the language of reinforcement learning

- **State**: Irregularity and degree of vertices in template
- **Action**: Flip, split, collapse, etc.
- **Reward**: $r_t = s_t - s_{t+1}$

Training procedure:

- Generate random 10-30 sided polygons
- Initial mesh by Delaunay refinement, split using Catmull-Clark for quads
- Terminate if $s^* = s$ or a maximum number of steps taken
- Monitor normalized returns

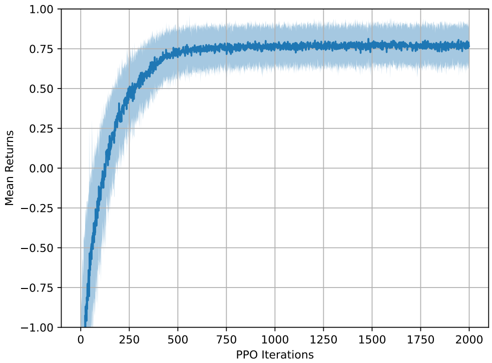
Neural network learns a mesh edit policy



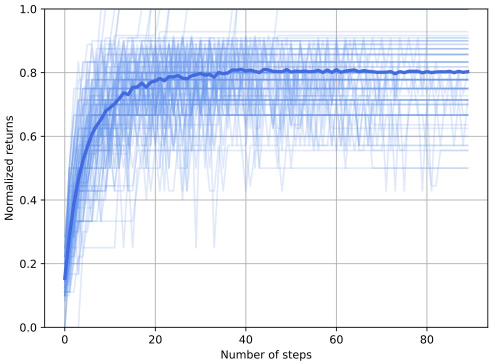
Trained in self-play by Proximal Policy Optimization (PPO) algorithm

Schulman, John, et al. *Proximal policy optimization algorithms* arXiv:1707.06347 (2017).

Results: Triangular Meshes



Average performance over training history



Evaluating the trained agent on multiple rollouts

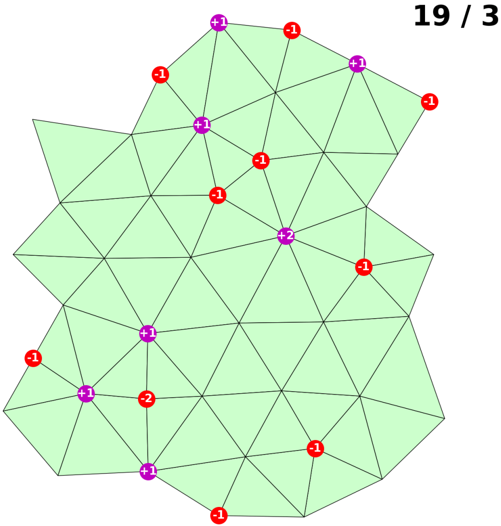
Performance of the triangle mesh agent over the training history.

Results: Triangular Meshing

Triangular meshing

Example 1

Step 0 (out of 27)

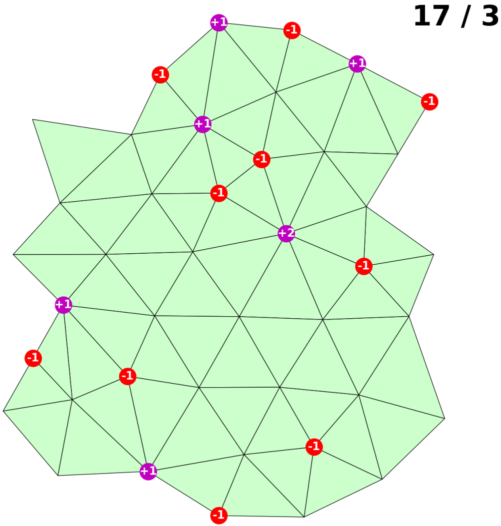


Results: Triangular Meshing

Triangular meshing

Example 1

Step 1 (out of 27)

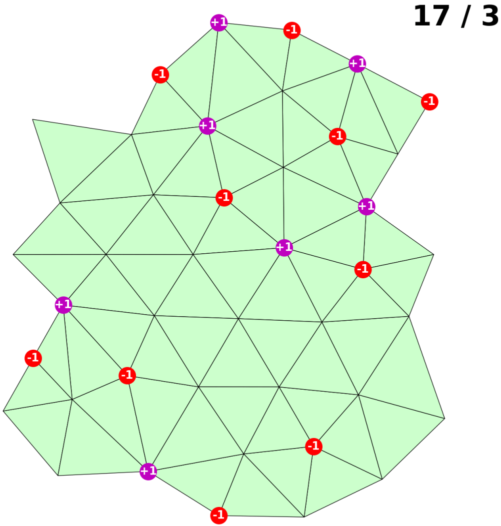


Results: Triangular Meshing

Triangular meshing

Example 1

Step 2 (out of 27)

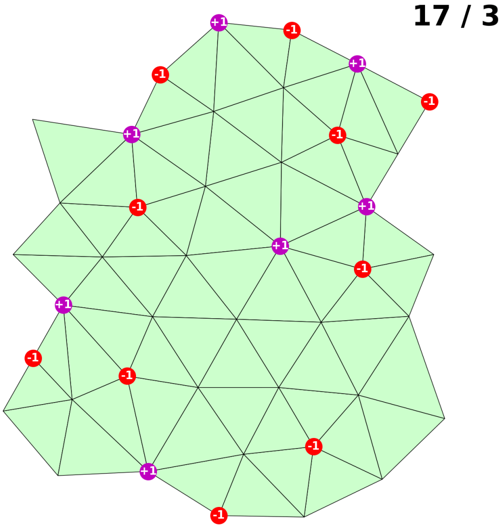


Results: Triangular Meshing

Triangular meshing

Example 1

Step 3 (out of 27)

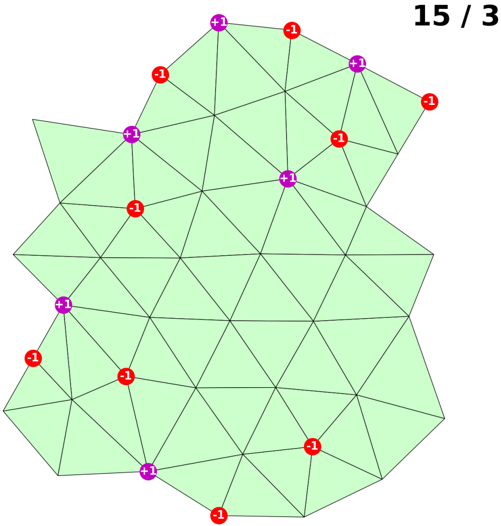


Results: Triangular Meshing

Triangular meshing

Example 1

Step 4 (out of 27)

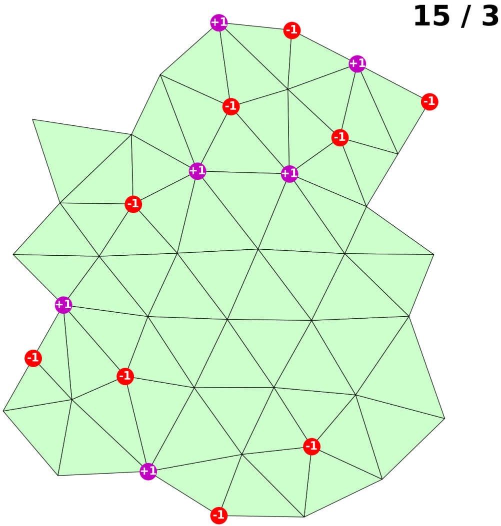


Results: Triangular Meshing

Triangular meshing

Example 1

Step 5 (out of 27)

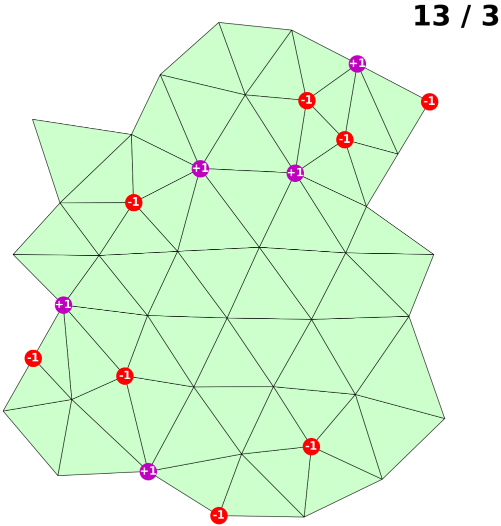


Results: Triangular Meshing

Triangular meshing

Example 1

Step 6 (out of 27)

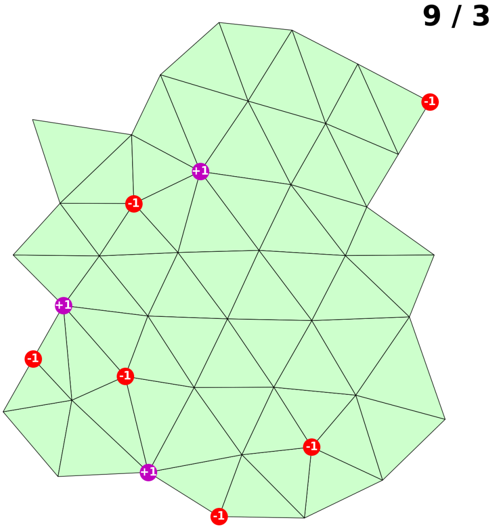


Results: Triangular Meshing

Triangular meshing

Example 1

Step 7 (out of 27)

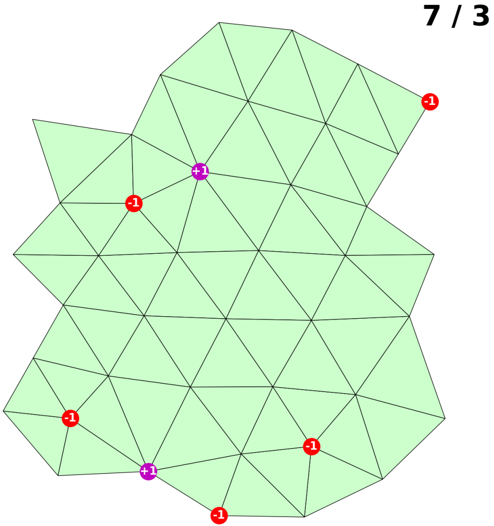


Results: Triangular Meshing

Triangular meshing

Example 1

Step 8 (out of 27)

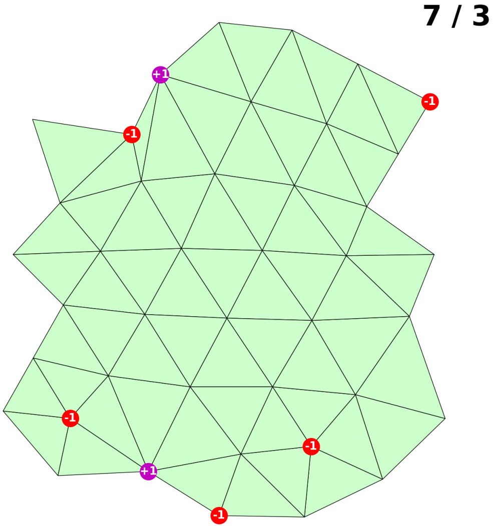


Results: Triangular Meshing

Triangular meshing

Example 1

Step 9 (out of 27)

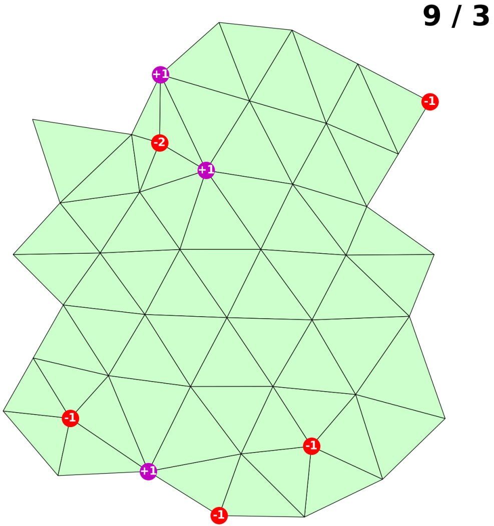


Results: Triangular Meshing

Triangular meshing

Example 1

Step 10 (out of 27)

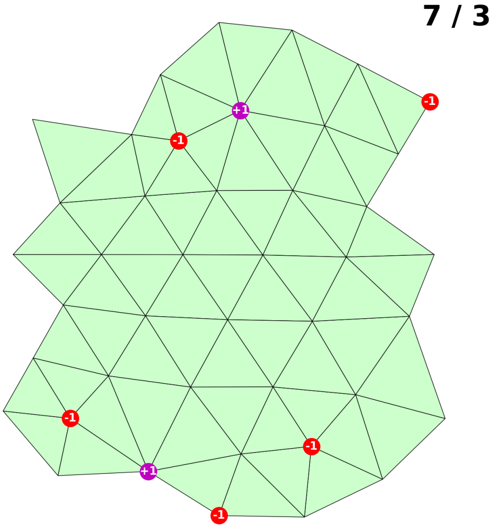


Results: Triangular Meshing

Triangular meshing

Example 1

Step 11 (out of 27)

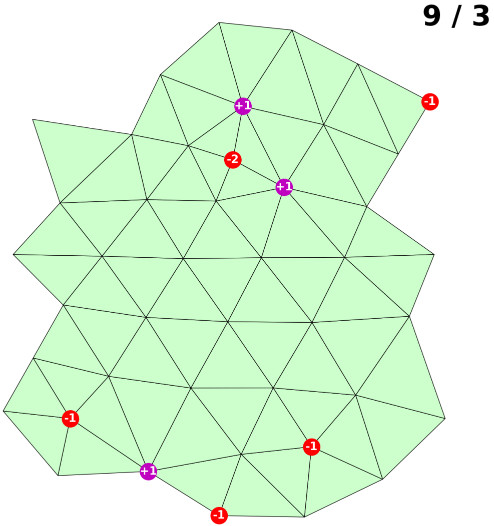


Results: Triangular Meshing

Triangular meshing

Example 1

Step 12 (out of 27)

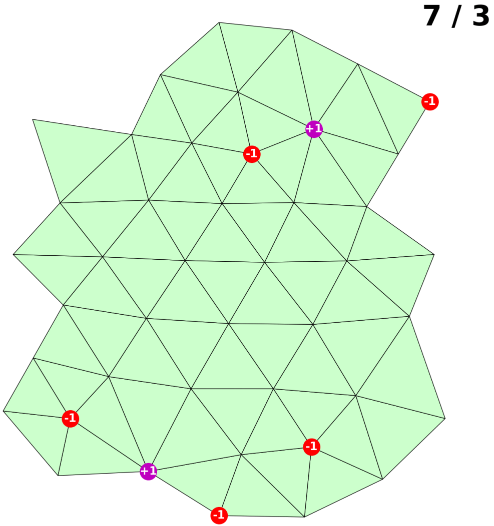


Results: Triangular Meshing

Triangular meshing

Example 1

Step 13 (out of 27)

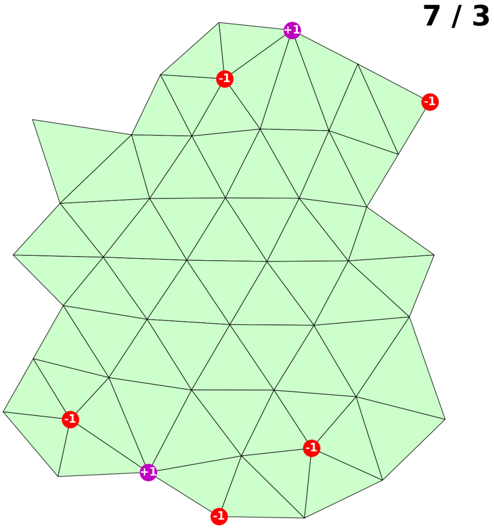


Results: Triangular Meshing

Triangular meshing

Example 1

Step 14 (out of 27)

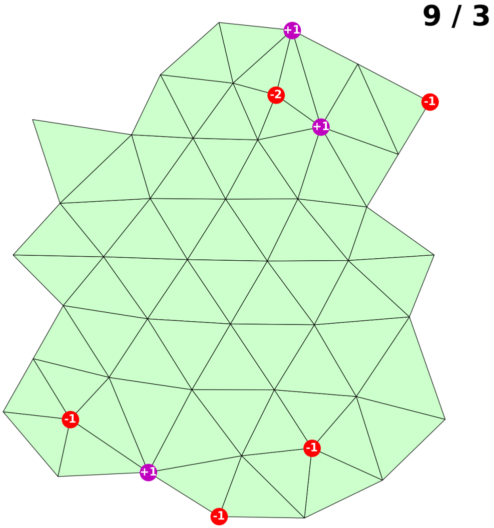


Results: Triangular Meshing

Triangular meshing

Example 1

Step 15 (out of 27)

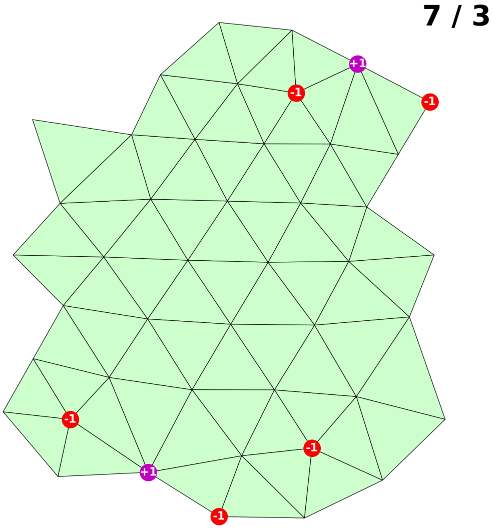


Results: Triangular Meshing

Triangular meshing

Example 1

Step 16 (out of 27)

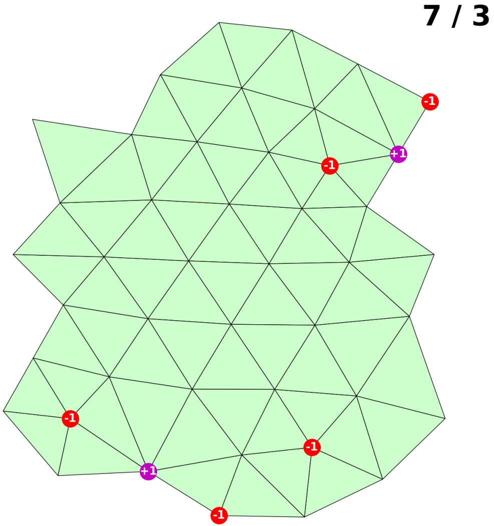


Results: Triangular Meshing

Triangular meshing

Example 1

Step 17 (out of 27)

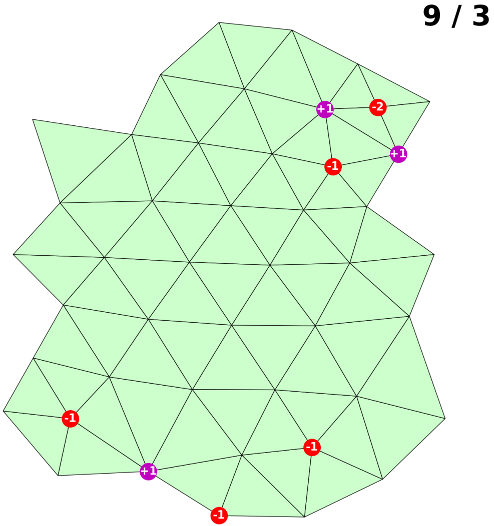


Results: Triangular Meshing

Triangular meshing

Example 1

Step 18 (out of 27)

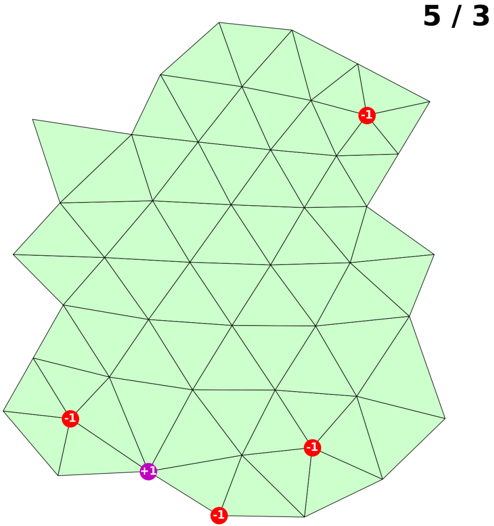


Results: Triangular Meshing

Triangular meshing

Example 1

Step 19 (out of 27)

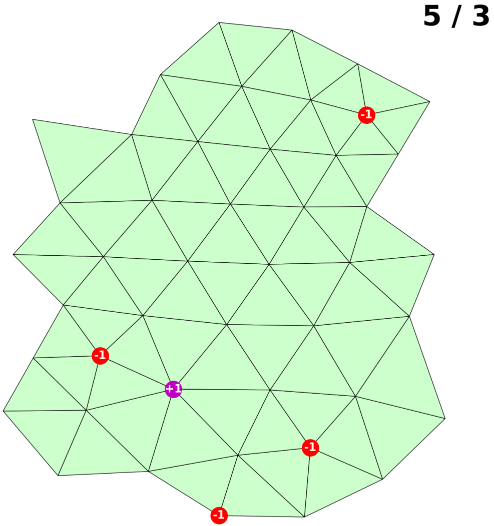


Results: Triangular Meshing

Triangular meshing

Example 1

Step 20 (out of 27)

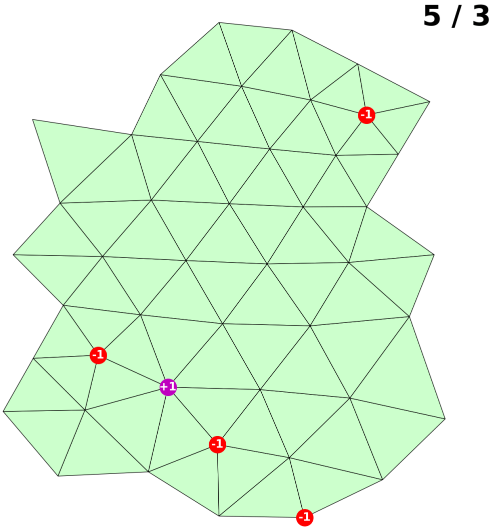


Results: Triangular Meshing

Triangular meshing

Example 1

Step 21 (out of 27)

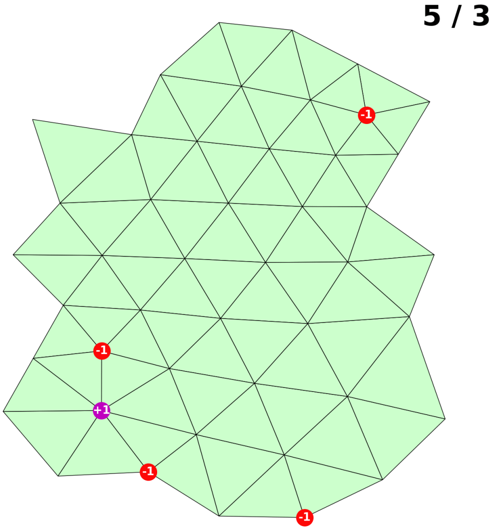


Results: Triangular Meshing

Triangular meshing

Example 1

Step 22 (out of 27)

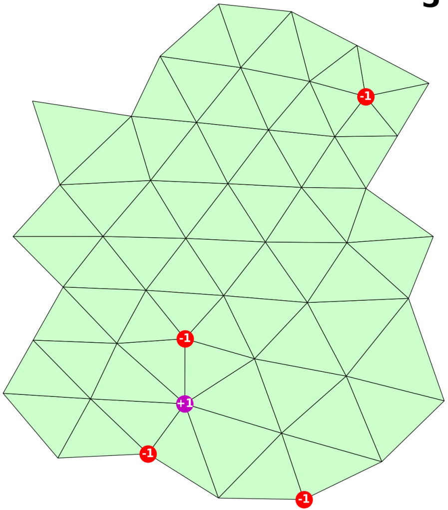


Results: Triangular Meshing

Triangular meshing

Example 1

Step 23 (out of 27)

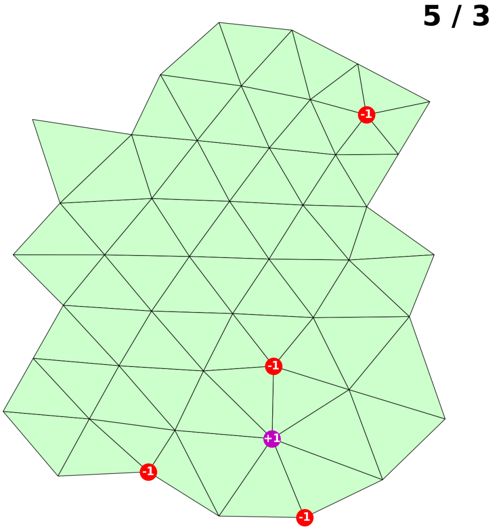


Results: Triangular Meshing

Triangular meshing

Example 1

Step 24 (out of 27)

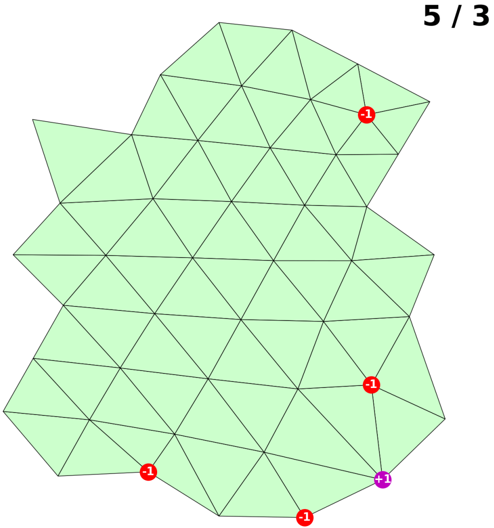


Results: Triangular Meshing

Triangular meshing

Example 1

Step 25 (out of 27)

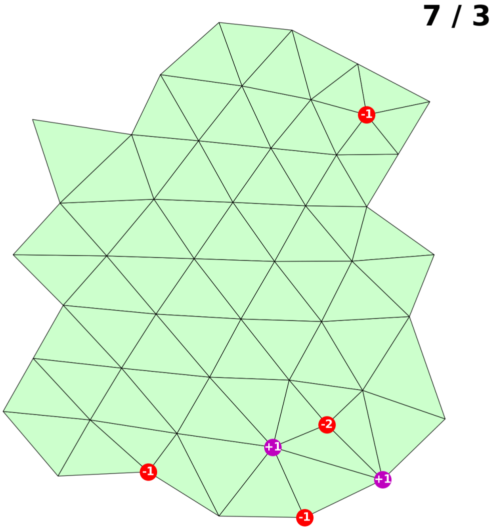


Results: Triangular Meshing

Triangular meshing

Example 1

Step 26 (out of 27)

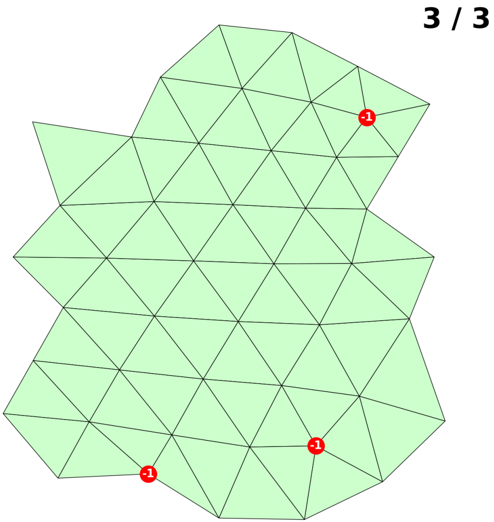


Results: Triangular Meshing

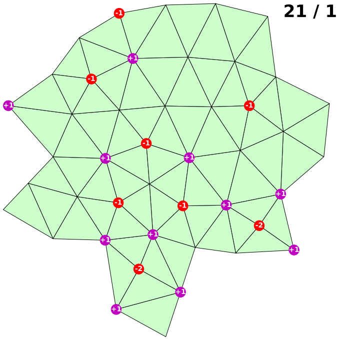
Triangular meshing

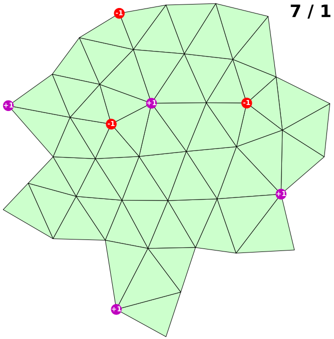
Example 1

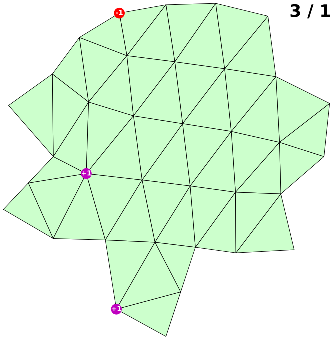
Step 27 (out of 27)



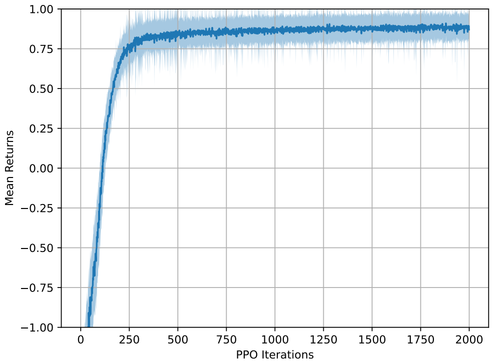
Triangular meshing example: 20-sided polygon



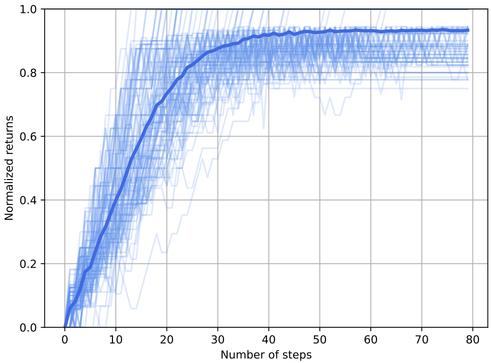




Results: Quadrilateral Meshes



Average performance over training history



Evaluating the trained agent on multiple rollouts

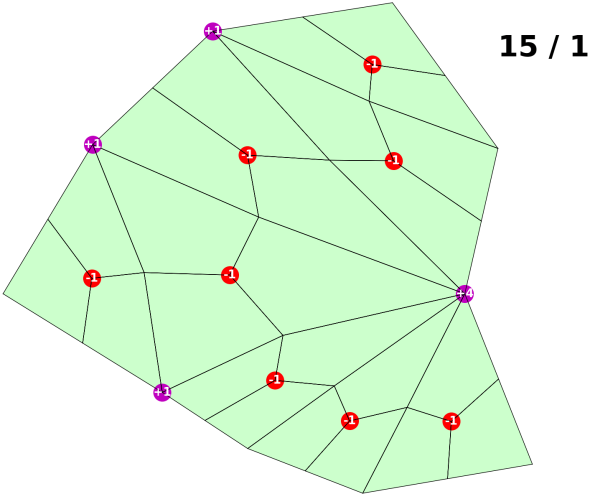
Performance of the quadrilateral mesh agent over the training history.

Results: Quadrilateral block meshing

Block mesh decomposition

Example 1

Step 0 (out of 19)

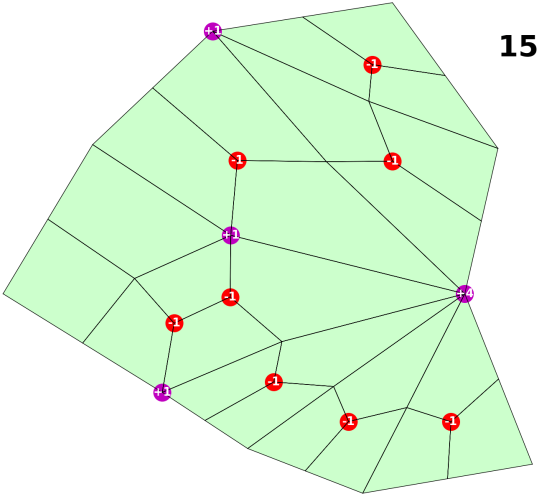


Results: Quadrilateral block meshing

Block mesh decomposition

Example 1

Step 1 (out of 19)



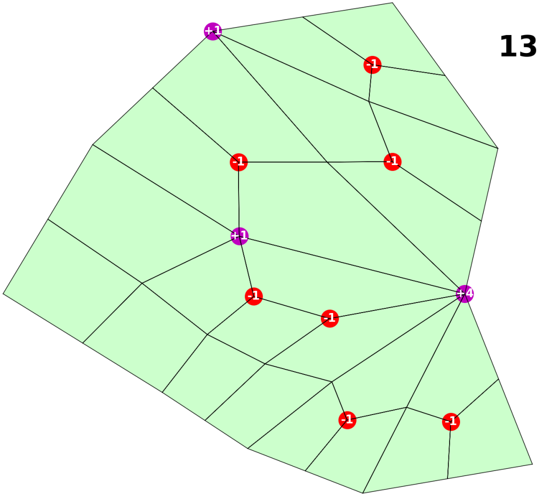
15 / 1

Results: Quadrilateral block meshing

Block mesh decomposition

Example 1

Step 2 (out of 19)



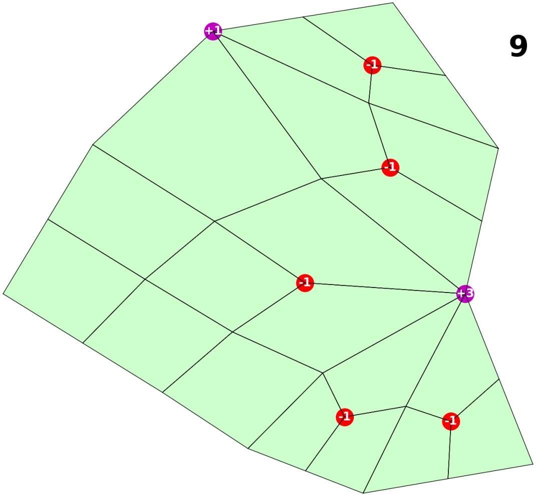
13 / 1

Results: Quadrilateral block meshing

Block mesh decomposition

Example 1

Step 3 (out of 19)



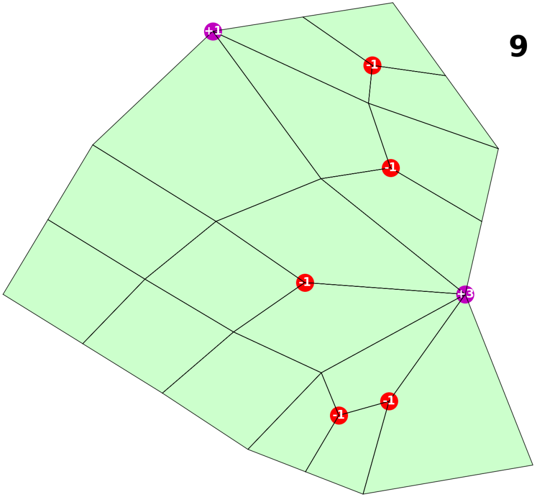
9 / 1

Results: Quadrilateral block meshing

Block mesh decomposition

Example 1

Step 4 (out of 19)



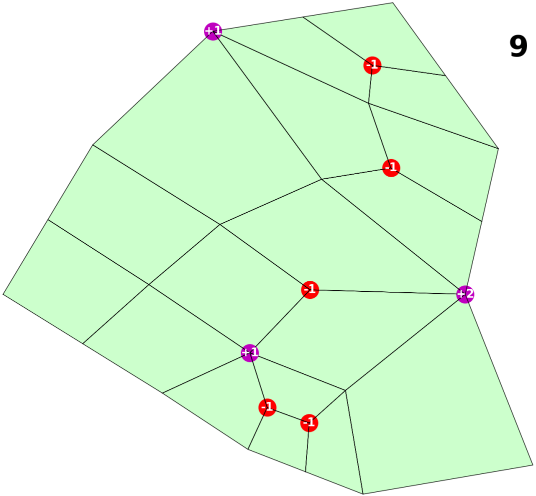
9 / 1

Results: Quadrilateral block meshing

Block mesh decomposition

Example 1

Step 5 (out of 19)



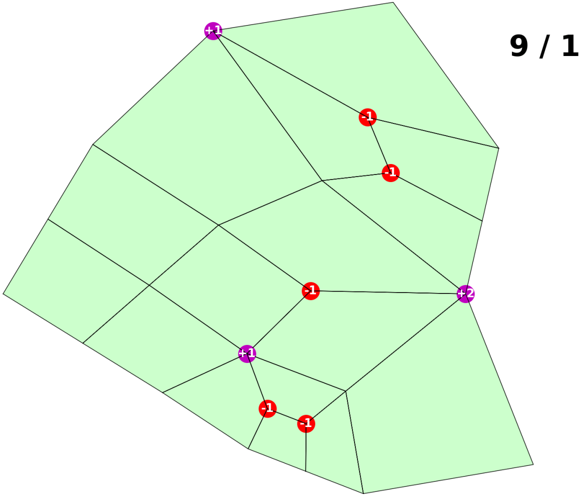
9 / 1

Results: Quadrilateral block meshing

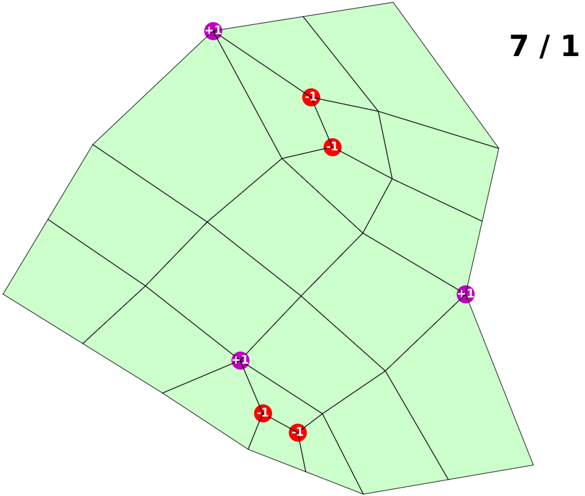
Block mesh decomposition

Example 1

Step 6 (out of 19)



Results: Quadrilateral block meshing



Block mesh decomposition

Example 1

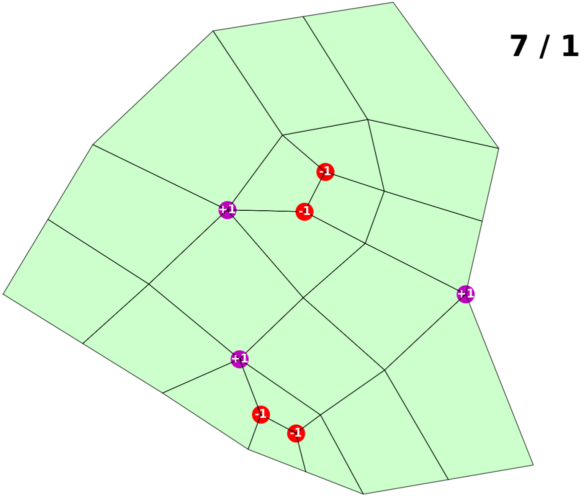
Step 7 (out of 19)

Results: Quadrilateral block meshing

Block mesh decomposition

Example 1

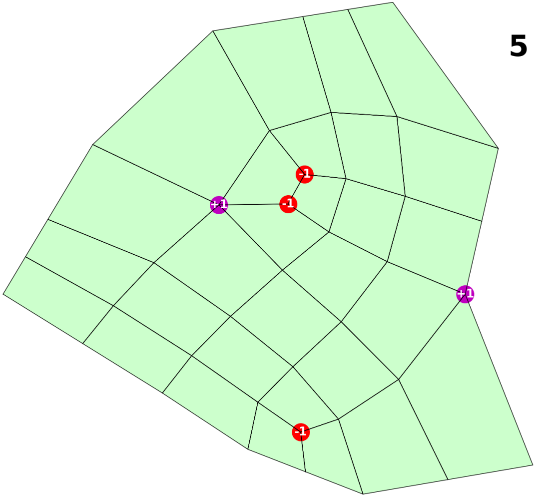
Step 8 (out of 19)



Block mesh decomposition

Example 1

Step 9 (out of 19)



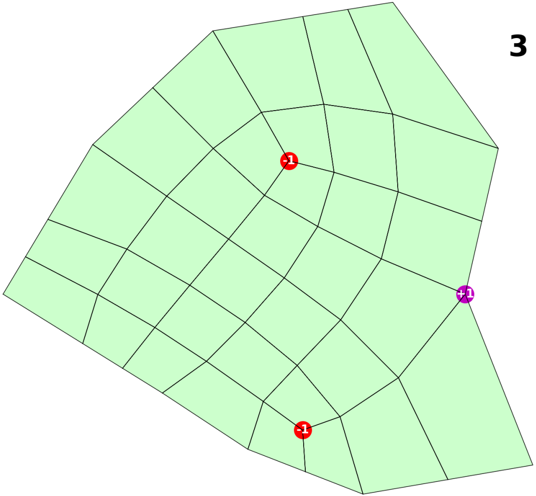
5 / 1

Results: Quadrilateral block meshing

Block mesh decomposition

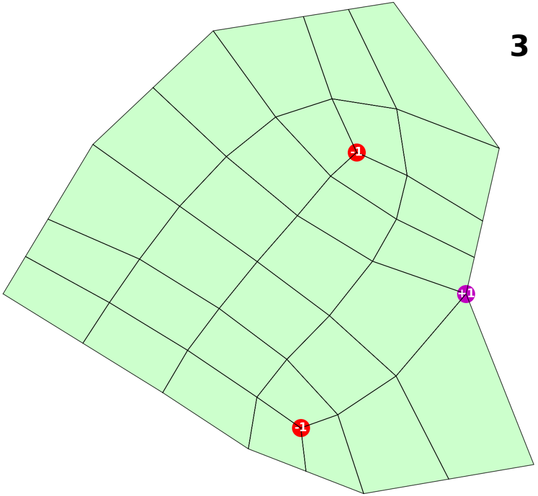
Example 1

Step 10 (out of 19)



3 / 1

Results: Quadrilateral block meshing



3 / 1

Block mesh decomposition

Example 1

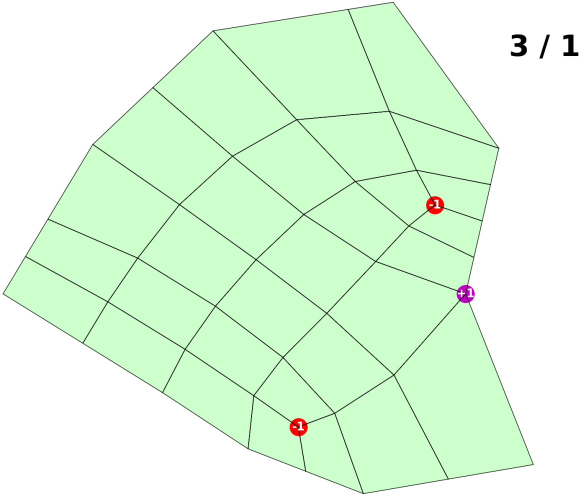
Step 11 (out of 19)

Results: Quadrilateral block meshing

Block mesh decomposition

Example 1

Step 12 (out of 19)

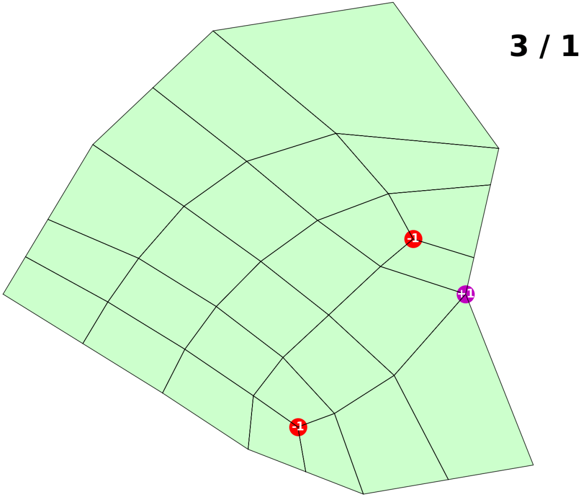


Results: Quadrilateral block meshing

Block mesh decomposition

Example 1

Step 13 (out of 19)

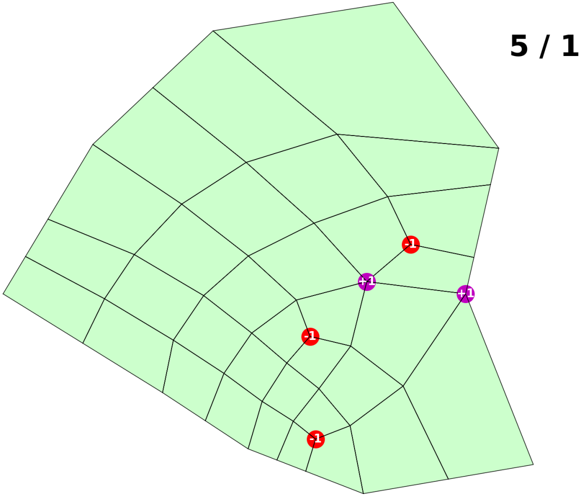


Results: Quadrilateral block meshing

Block mesh decomposition

Example 1

Step 14 (out of 19)

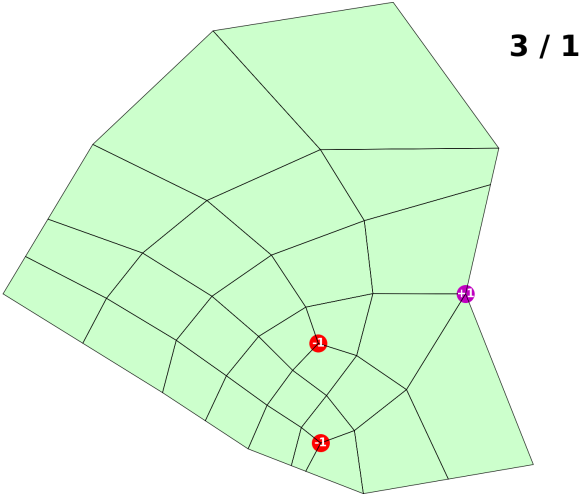


Results: Quadrilateral block meshing

Block mesh decomposition

Example 1

Step 15 (out of 19)

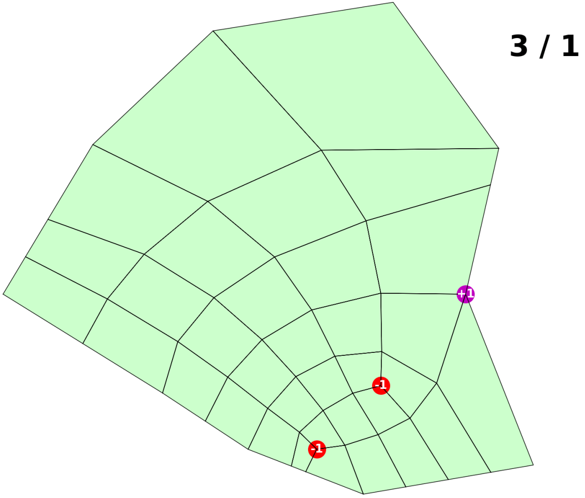


Results: Quadrilateral block meshing

Block mesh decomposition

Example 1

Step 16 (out of 19)

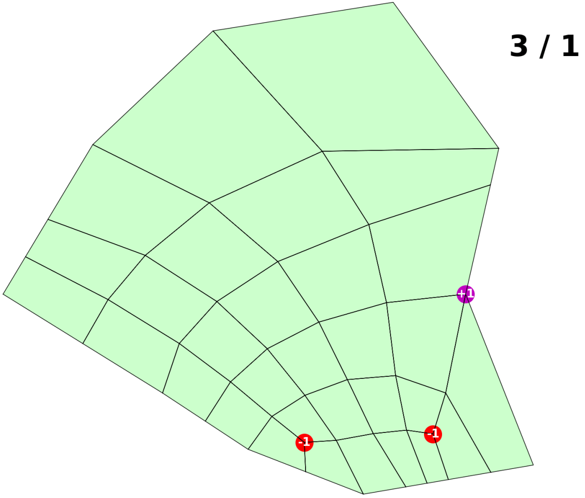


Results: Quadrilateral block meshing

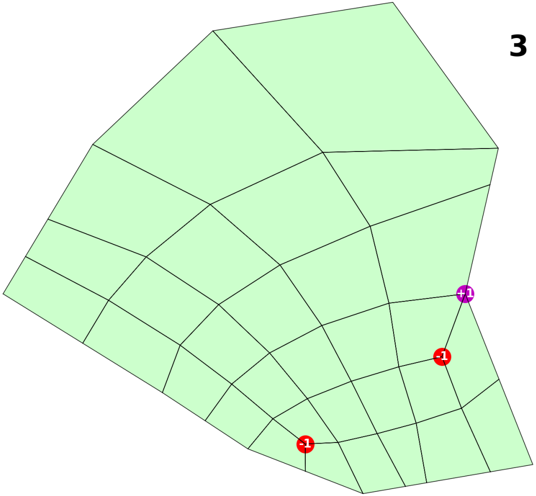
Block mesh decomposition

Example 1

Step 17 (out of 19)



Results: Quadrilateral block meshing



Block mesh decomposition

Example 1

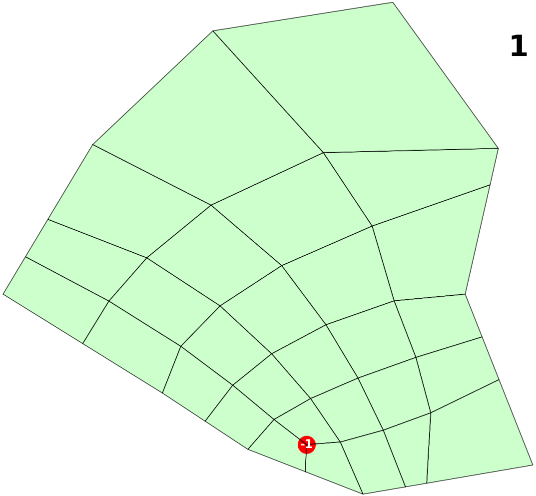
Step 18 (out of 19)

Results: Quadrilateral block meshing

Block mesh decomposition

Example 1

Step 19 (out of 19)



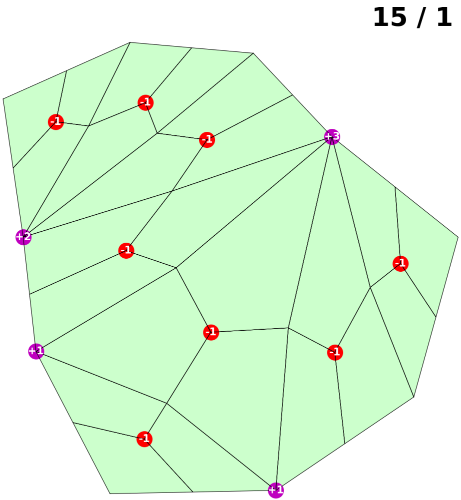
1 / 1

Results: Quadrilateral block meshing

Block mesh decomposition

Example 2

Step 0 (out of 12)

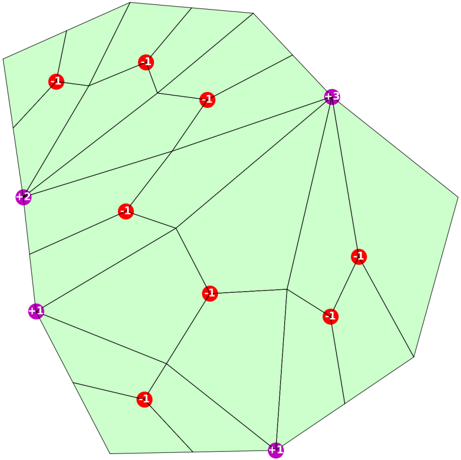


Block mesh decomposition

Example 2

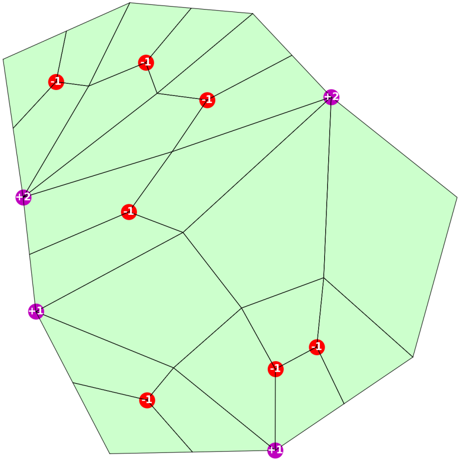
Step 1 (out of 12)

15 / 1



Results: Quadrilateral block meshing

13 / 1



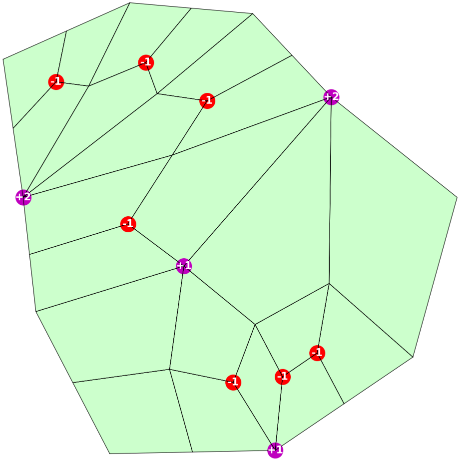
Block mesh decomposition

Example 2

Step 2 (out of 12)

Results: Quadrilateral block meshing

13 / 1



Block mesh decomposition

Example 2

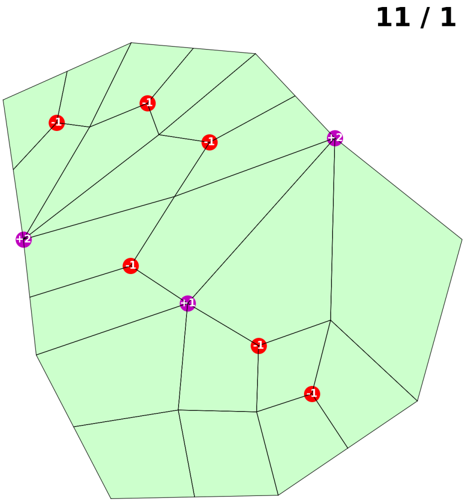
Step 3 (out of 12)

Results: Quadrilateral block meshing

Block mesh decomposition

Example 2

Step 4 (out of 12)

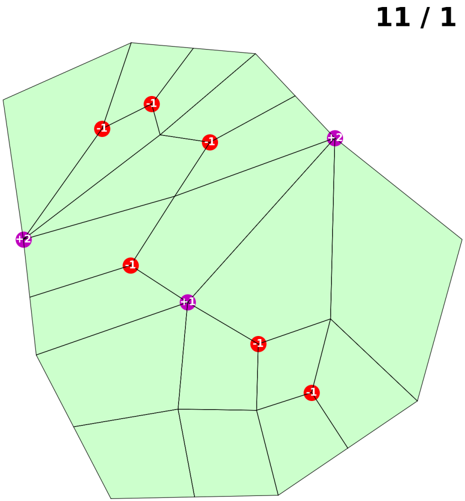


Results: Quadrilateral block meshing

Block mesh decomposition

Example 2

Step 5 (out of 12)

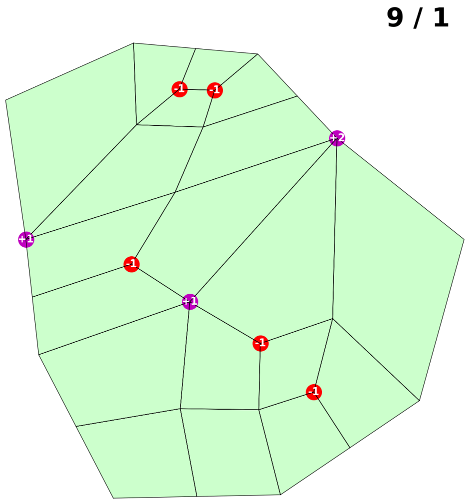


Results: Quadrilateral block meshing

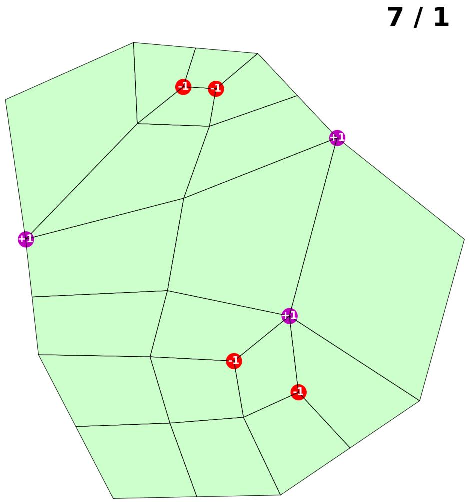
Block mesh decomposition

Example 2

Step 6 (out of 12)



Results: Quadrilateral block meshing



Block mesh decomposition

Example 2

Step 7 (out of 12)

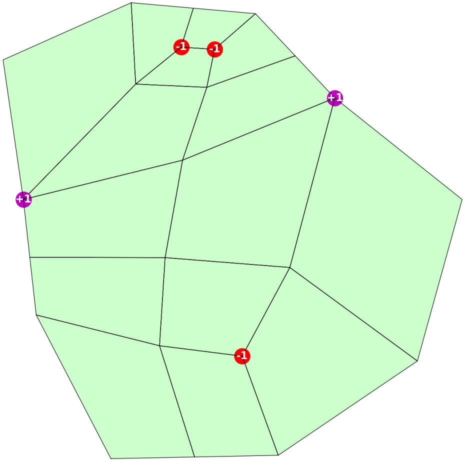
Results: Quadrilateral block meshing

Block mesh decomposition

Example 2

Step 8 (out of 12)

5 / 1

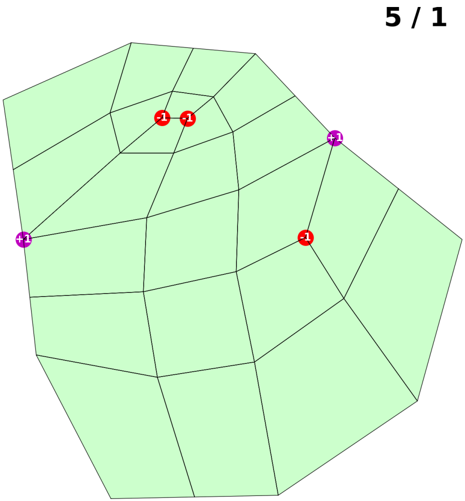


Results: Quadrilateral block meshing

Block mesh decomposition

Example 2

Step 9 (out of 12)

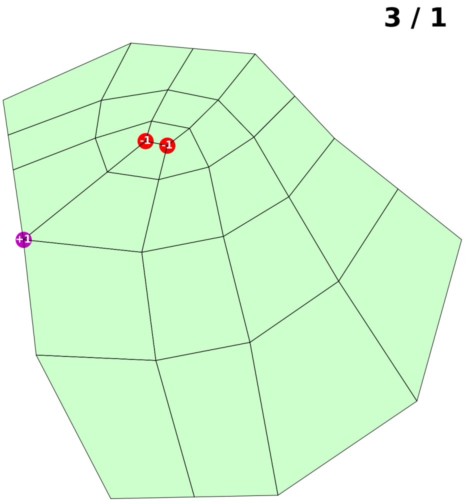


Results: Quadrilateral block meshing

Block mesh decomposition

Example 2

Step 10 (out of 12)

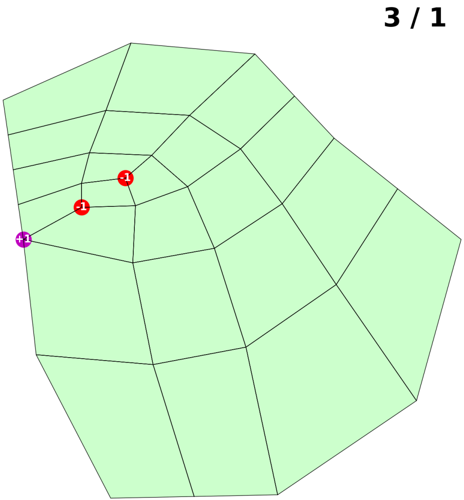


Results: Quadrilateral block meshing

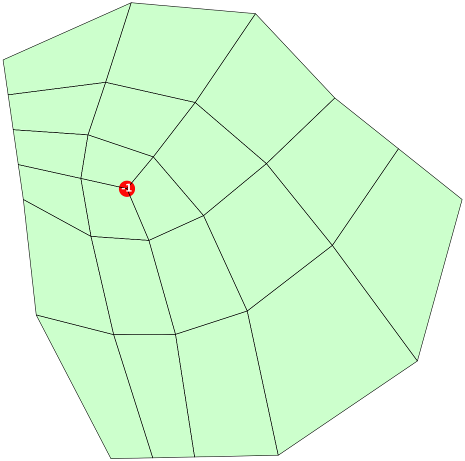
Block mesh decomposition

Example 2

Step 11 (out of 12)



Results: Quadrilateral block meshing

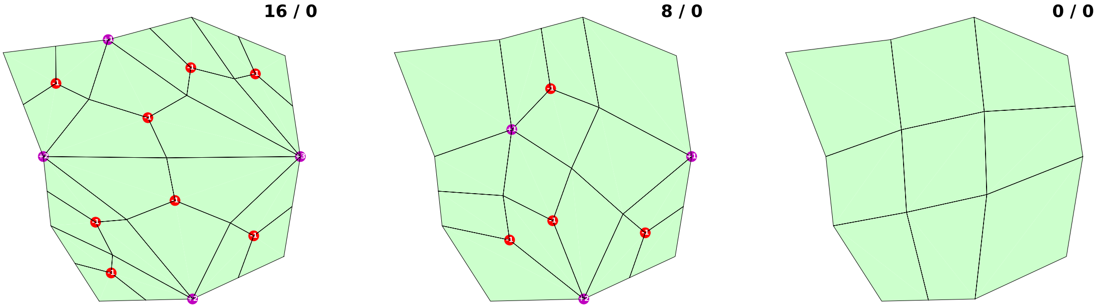


Block mesh decomposition

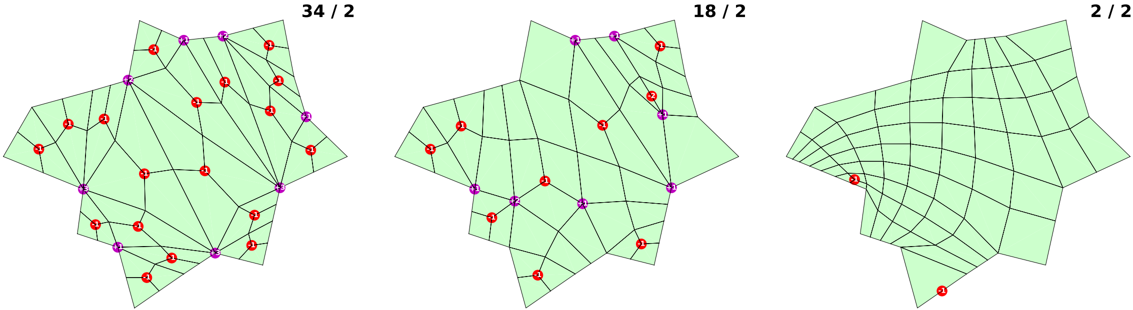
Example 2

Step 12 (out of 12)

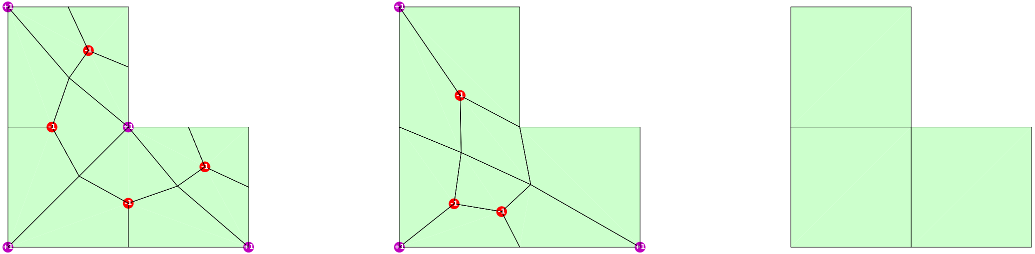
Block decomposition example: 10-sided polygon



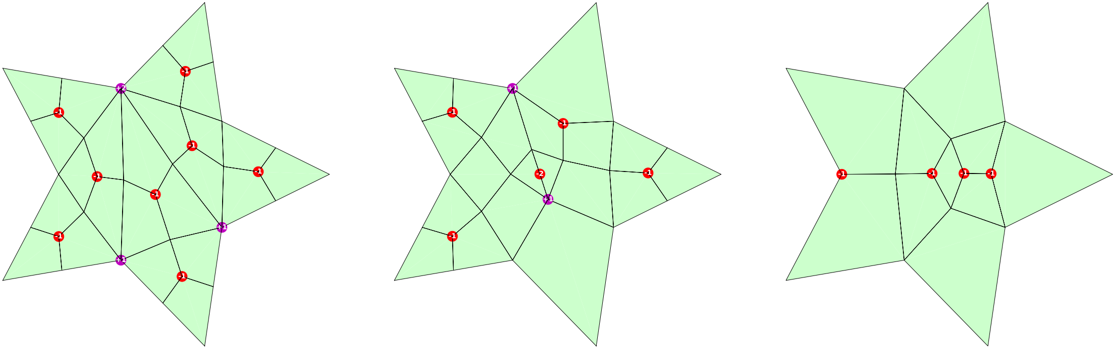
Block decomposition example: 20-sided polygon



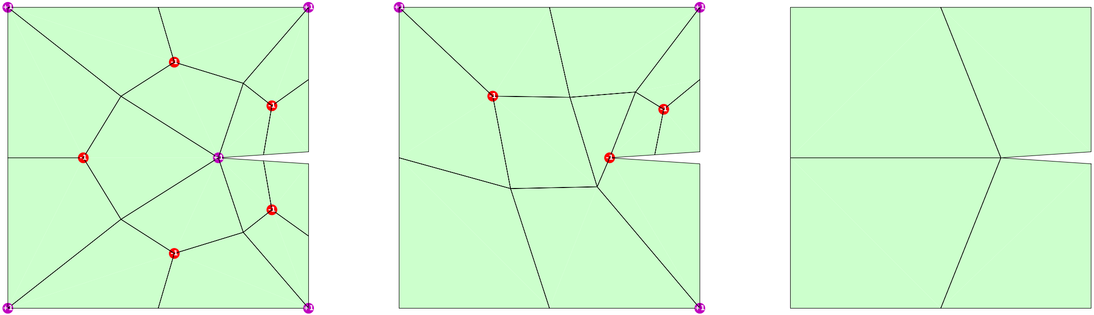
Block decomposition example: L-shaped domain



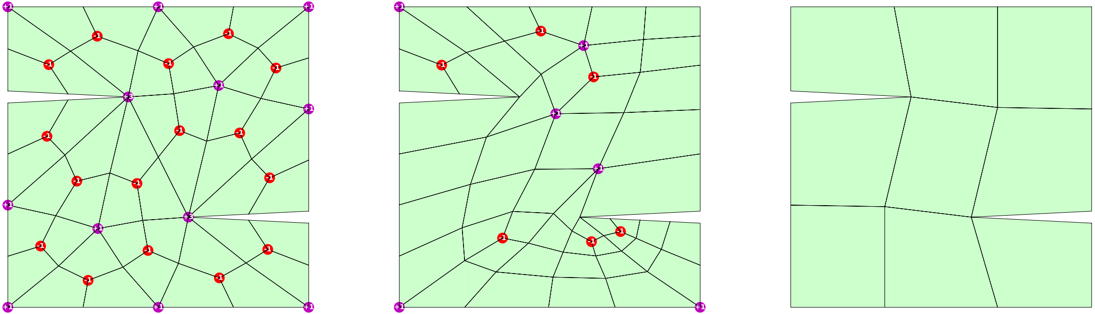
Block decomposition example: Star-shaped domain



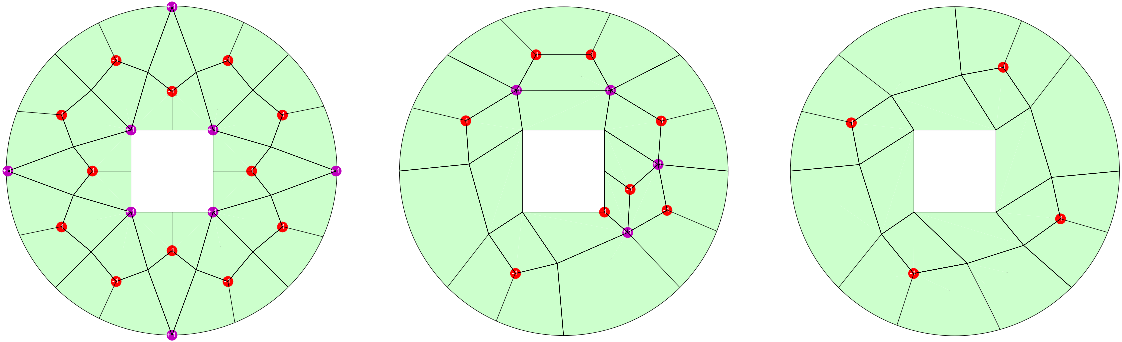
Block decomposition example: Notch domain



Block decomposition example: Double notch domain



Block decomposition example: Square hole in circle domain



Conclusions

- Representation of mesh topology for neural networks
- Unified method to optimize connectivity of triangular and quadrilateral meshes
- Heuristics-free method that learns rich behavior from self-play
- Future work: Combine with Monte Carlo Tree Search, more complex geometries, different formulations, optimize for element quality, 3D

[1] Narayanan, Pan, Persson. *Learning topological operations on meshes with application to block decomposition of polygons*. In review & arXiv:2309.06484.