Washington University in St. Louis
JAMES MCKELVEY SCHOOL OF ENGINEERING

TOPOLOGICAL SIMPLIFICATION OF NESTED SHAPES

Dan Zeng, Tao Ju (Washington University in St. Louis)

Erin Chambers, David Letscher (St. Louis University)

SGP 2022

Nested Shapes

Washington University in St. Louis
James McKelvey School of Engineering

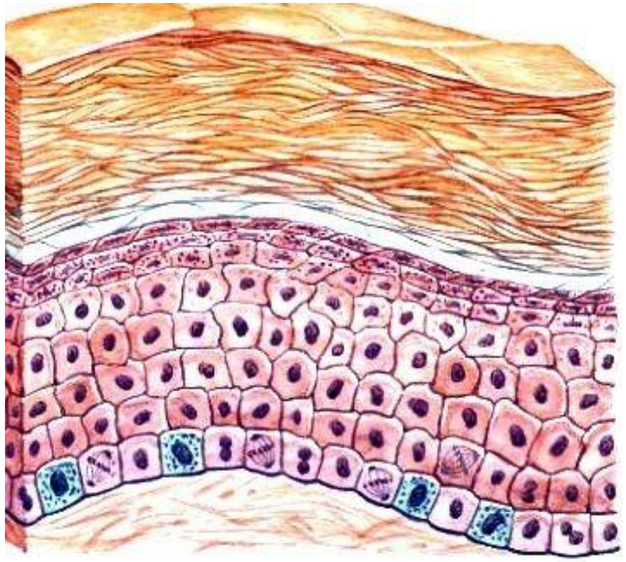
- A sequence of monotonically expanding shapes

Wikipedia

Nested Shapes

- Multi-layered structures

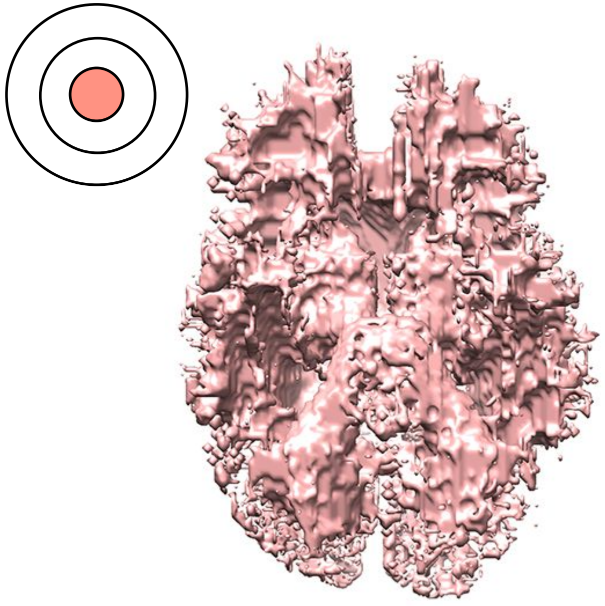
Geological strata



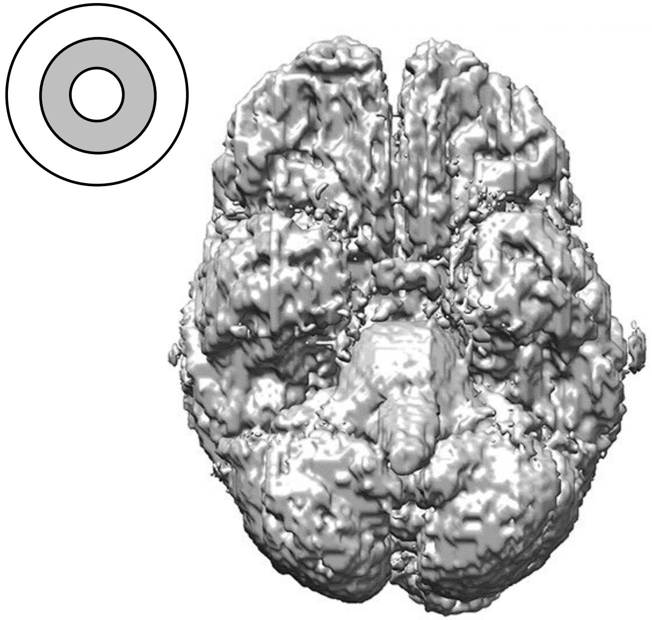
Tissue layers

Nested Shapes

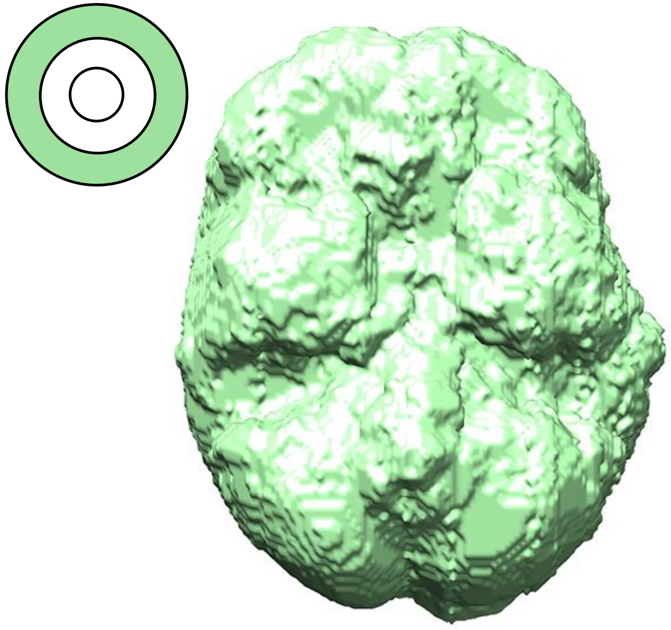
- Multi-layered structures



Cerebrospinal fluid



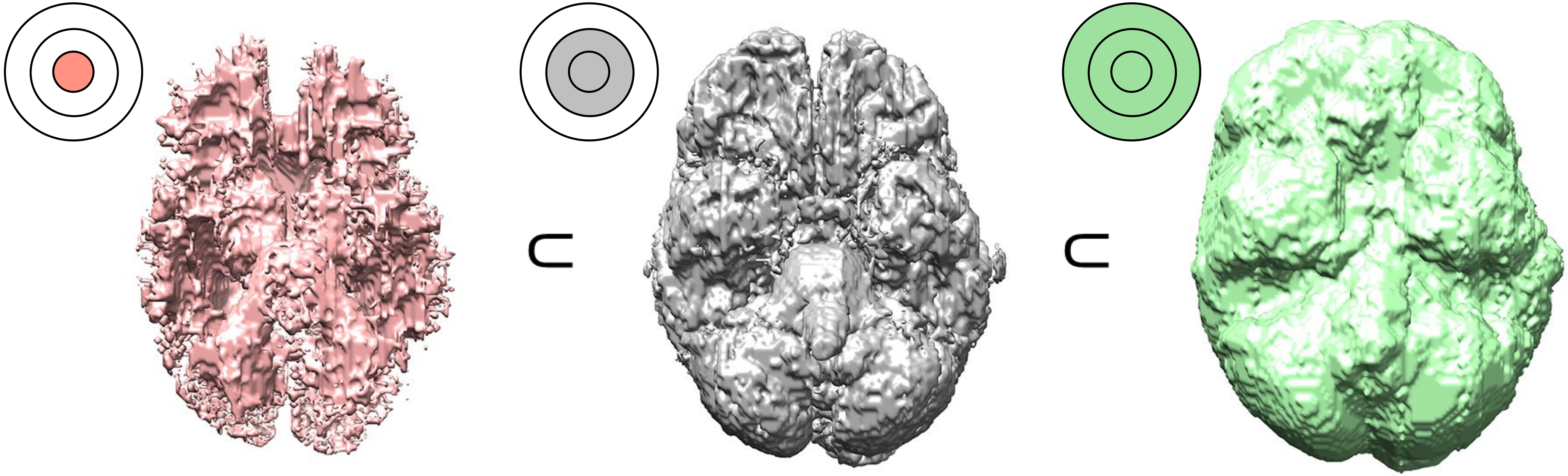
White matter



Gray matter

Nested Shapes

- Multi-layered structures
  - The outer surface of each layer forms a nested sequence



Cerebrospinal fluid ⊂ White matter ⊂ Gray matter

Nested Shapes

Washington University in St. Louis
James McKelvey School of Engineering

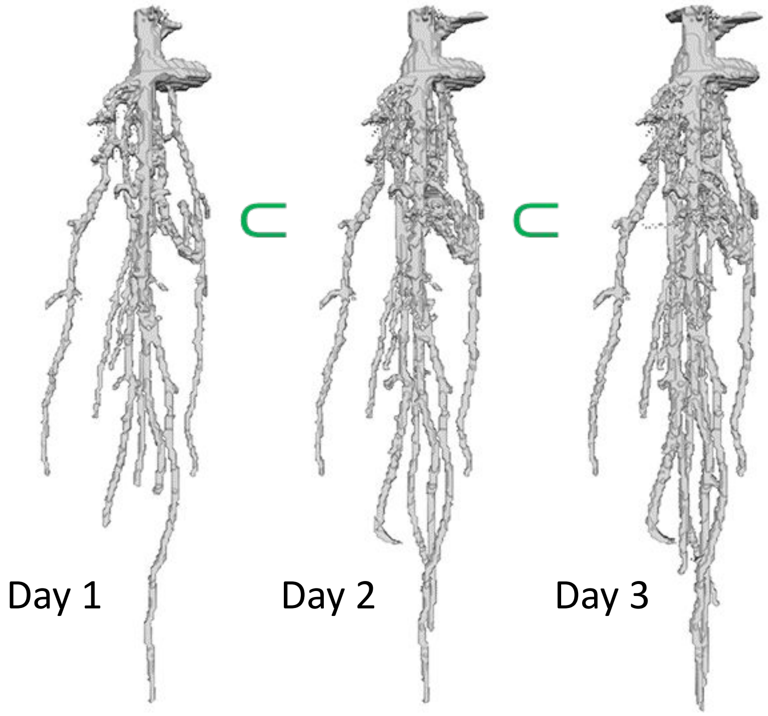
- Growing plant roots

Wikipedia

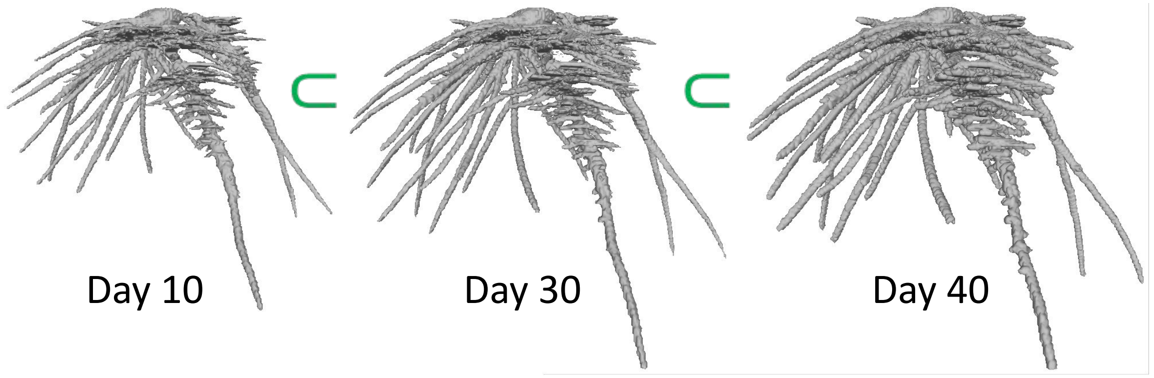
Nested Shapes

- Growing plant roots

Rice roots



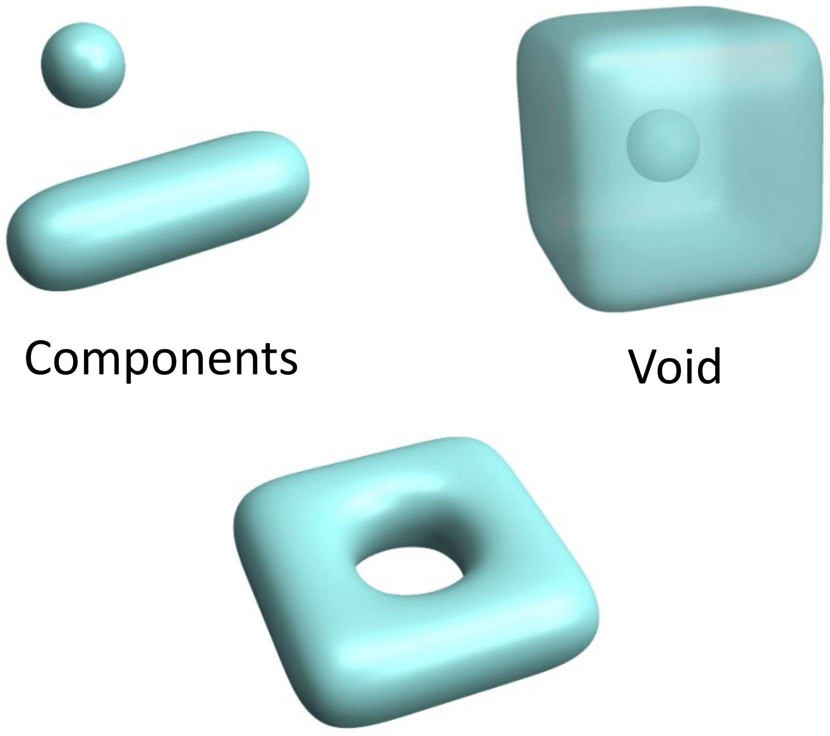
Maize (corn) roots



Topology

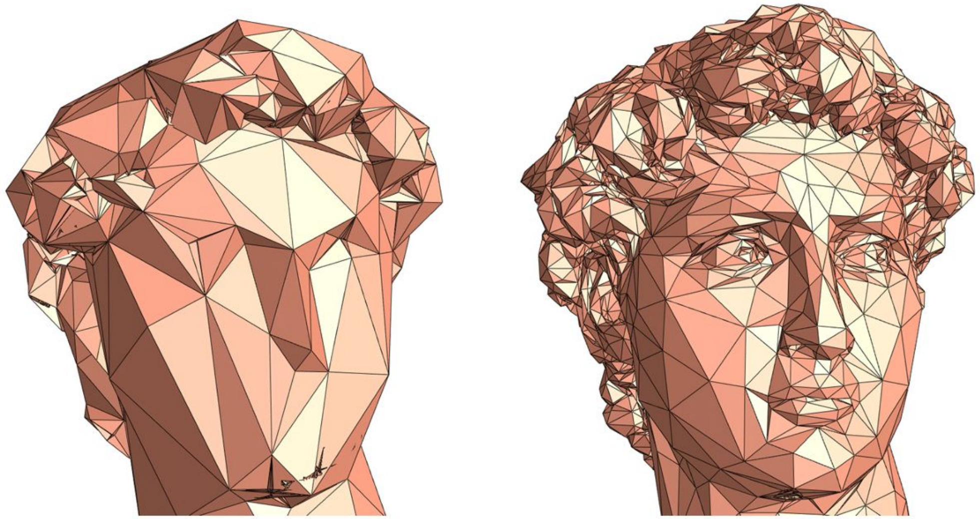
- Invariant to continuous geometric deformation

Wikipedia



Topology

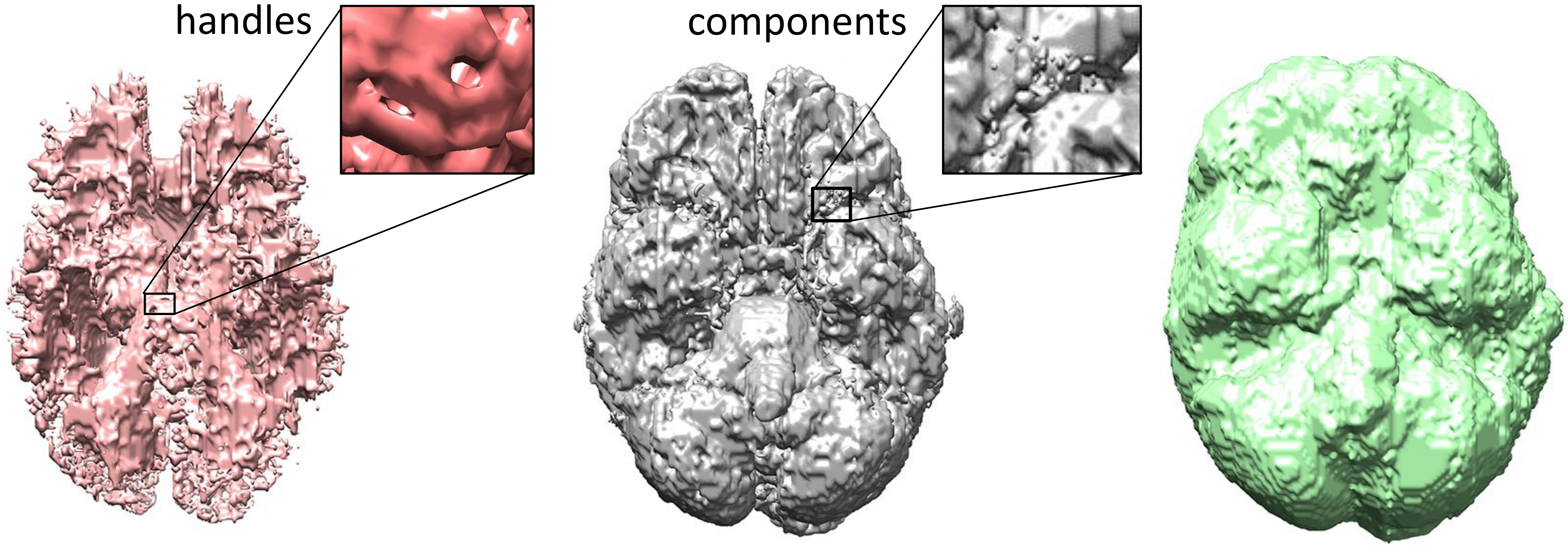
- Invariant to continuous geometric deformation
- Many geometry processing tasks are sensitive to topology:
  - Mesh simplification and fairing
  - Surface parameterization
  - Geodesic distances
  - Surface matching
  - Physical simulations



Mesh simplification before and after removing redundant topological features [Wood 04]

Topological Errors

- Reconstruction may introduce unwanted topological features

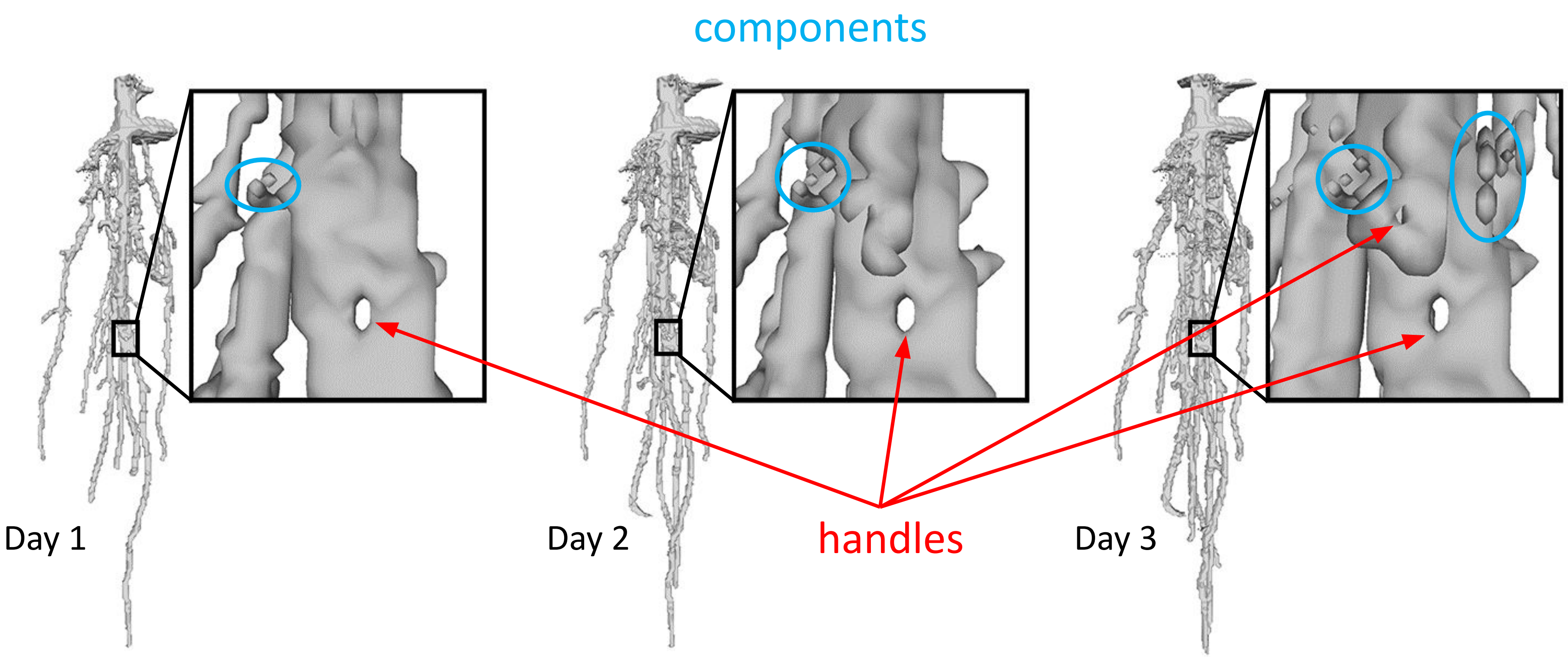


Cerebrospinal fluid White matter Gray matter

Topological Errors

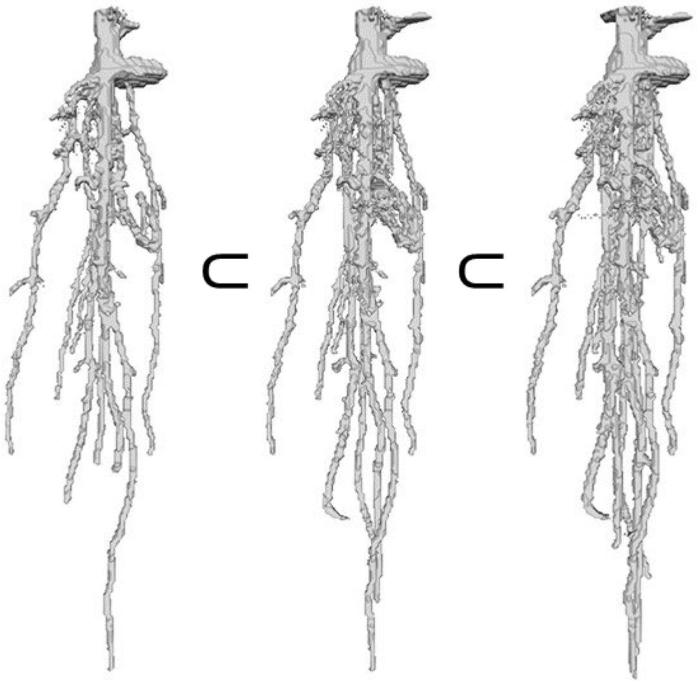
Washington University in St. Louis
James McKelvey School of Engineering

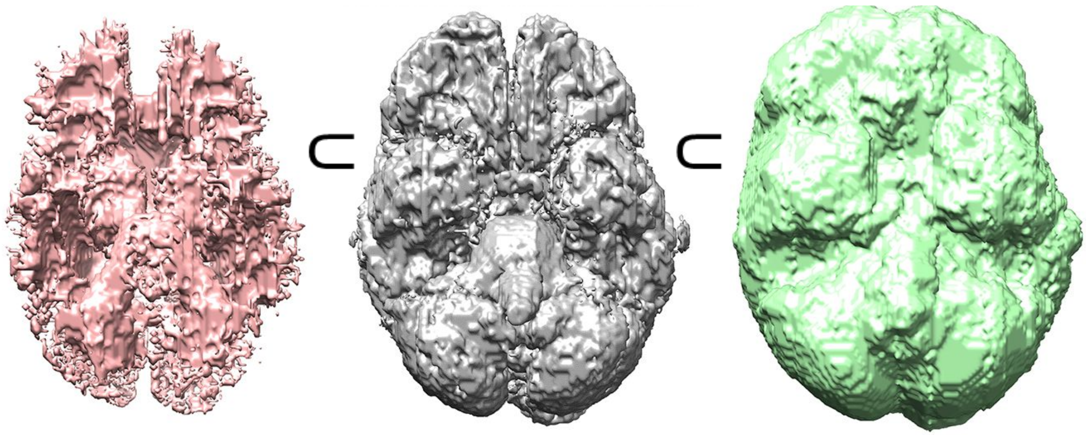
- Reconstruction may introduce unwanted topological features



Goal

- Remove unwanted topological features in reconstructed shapes
  - Maintain nesting (necessary for defining layers or modeling root growth)

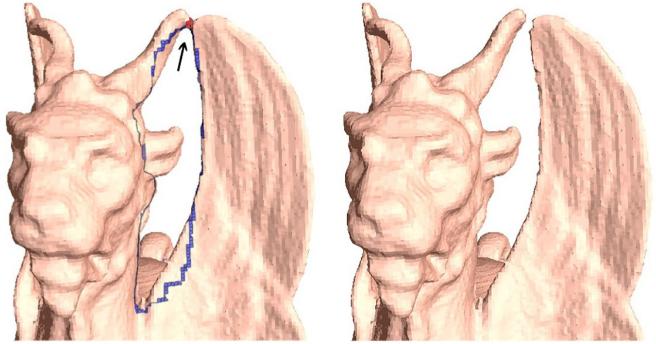




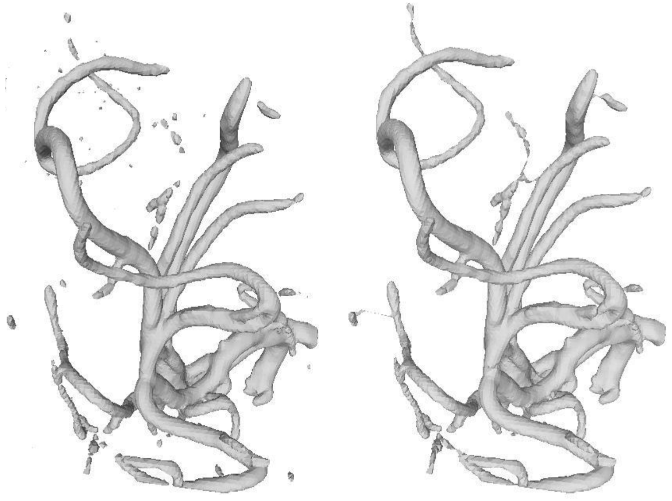
Previous Works

Washington University in St.Louis
James McKelvey School of Engineering

- Simplifying the topology of one shape
  - Removing handles [Shattuck 01; Han 02; Wood 04; Chen 06; Zhou 07; Segonne 07]
  - Removing all features
    - Morphological opening/closing [Nooruddin 03]
    - Inflation and deflation [Kriegeskorte 01; Bischoff 02; Szymczak 03]
    - Local heuristics [Ju 07]
    - Global optimization [Zeng 20]



[Wood 04]



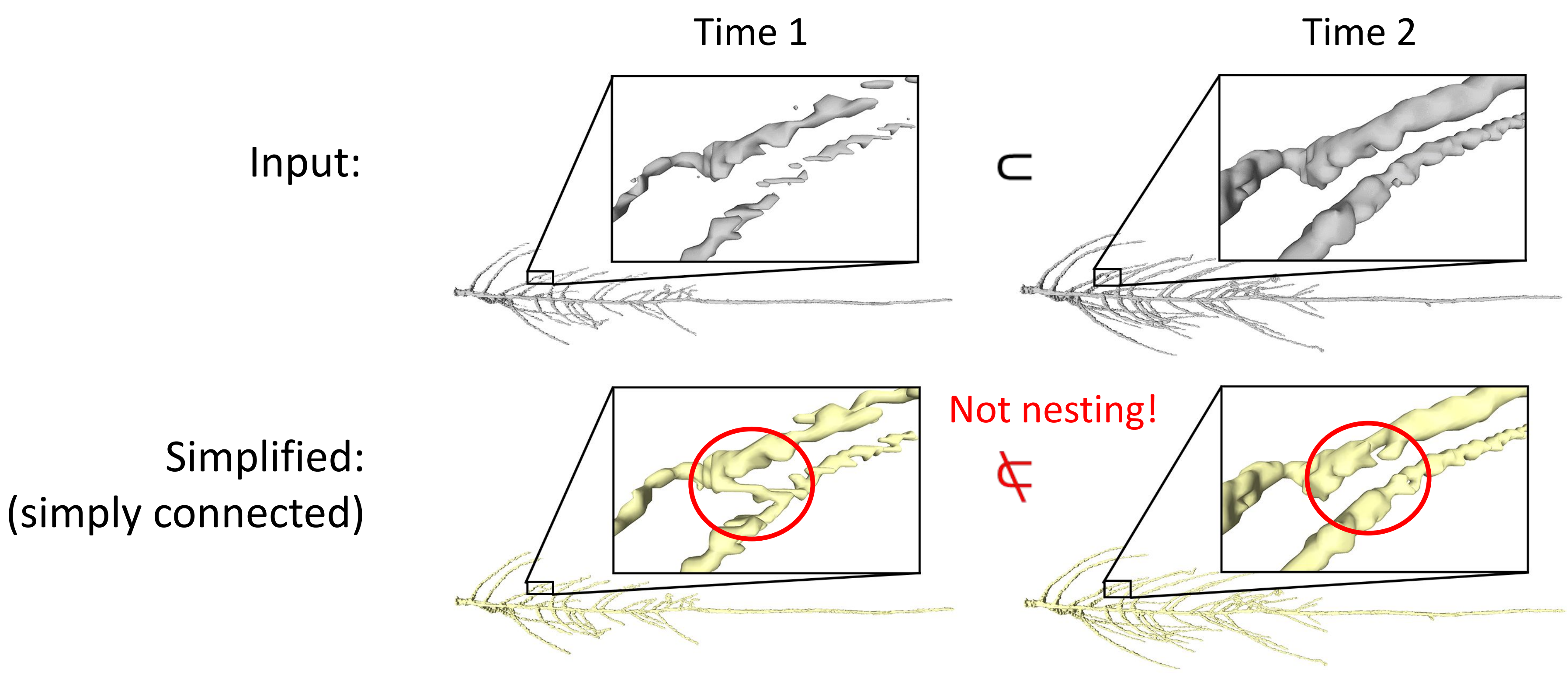
Input

Simply connected

[Zeng 20]

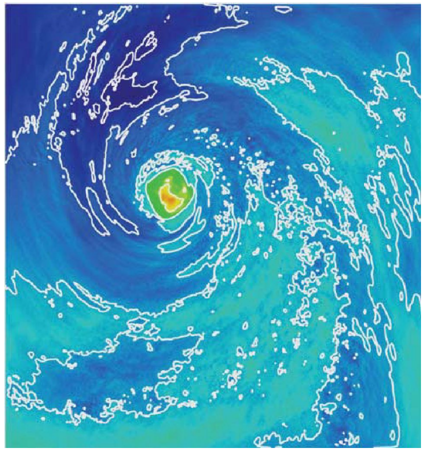
Previous Works

- Simplifying the topology of one shape
  - Cannot guarantee nesting when applied independently to each shape



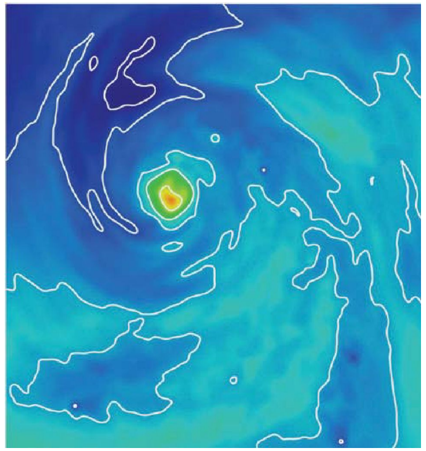
Previous Works

- Simplifying the topology of a scalar function
  - Removes extraneous critical points, thus simplifying the topology of *all* level sets (which are nested)
  - Numerical optimization [Bremer 04; Patane 09; Weinkauf 10; Gunther 14]
  - Combinatorial methods [Edelsbrunner 06; Bauer 12; Tierny 12,17; Lukasczyk 20]



14492 extrema

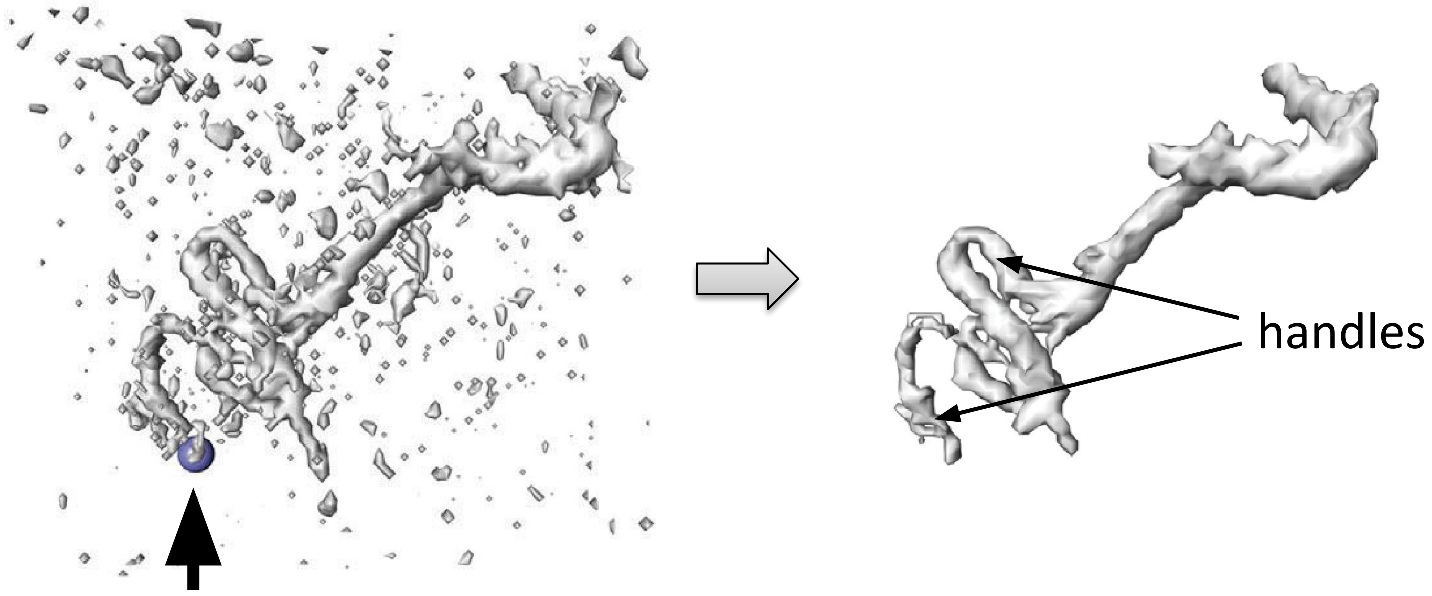
[Gunther 14]



12 extrema

Previous Works

- Simplifying the topology of a scalar function
  - Saddles in 3D (corresponding to handles of the level sets) are challenging to remove



Removing all local minima except one [Gunther 14]

Our Work

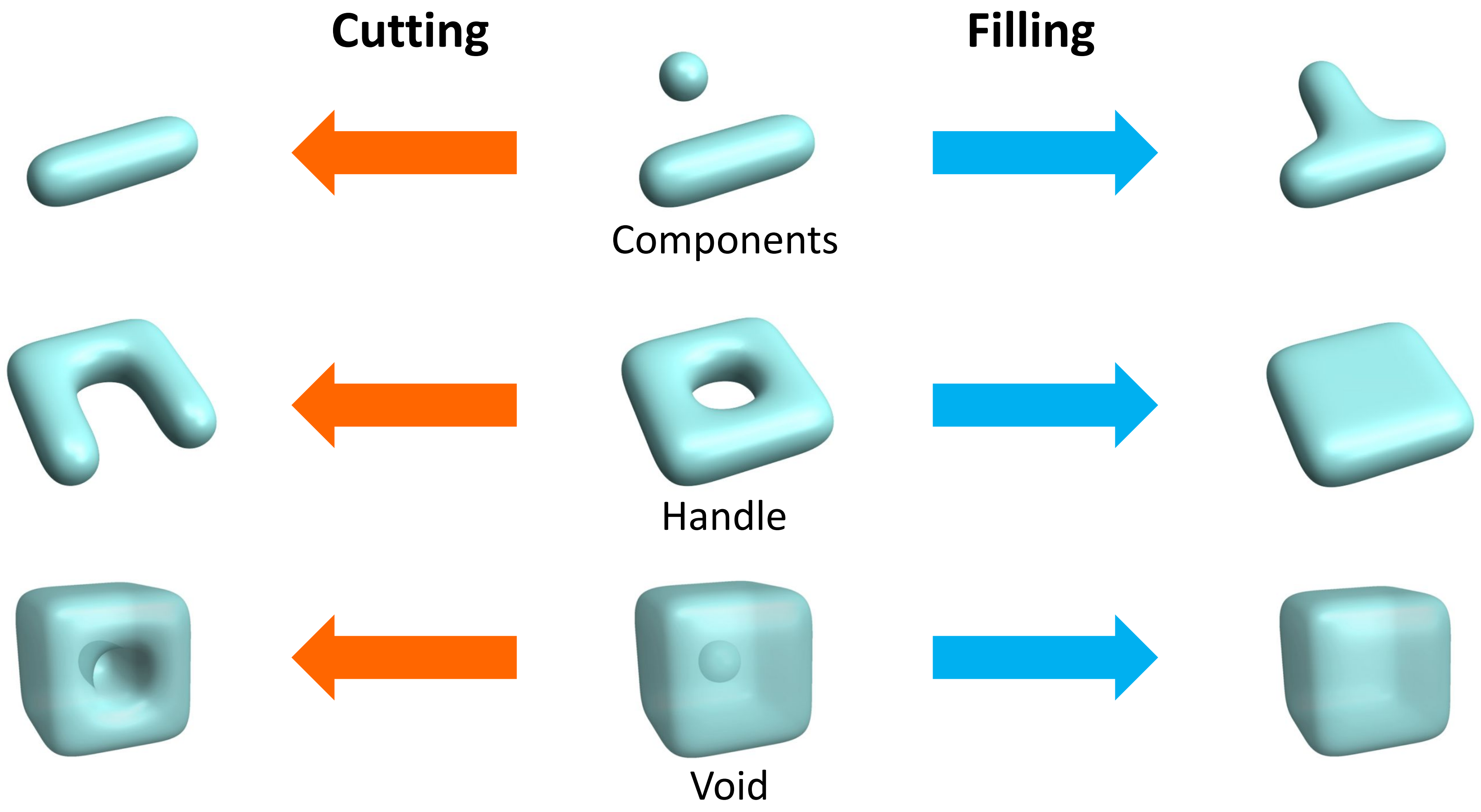
Washington University in St. Louis
James McKelvey School of Engineering

- Simplifies the topology of a shape sequence while maintaining nesting
  - Removes all three types of topological features (components, handles, voids)
  - Minimally alters the shapes

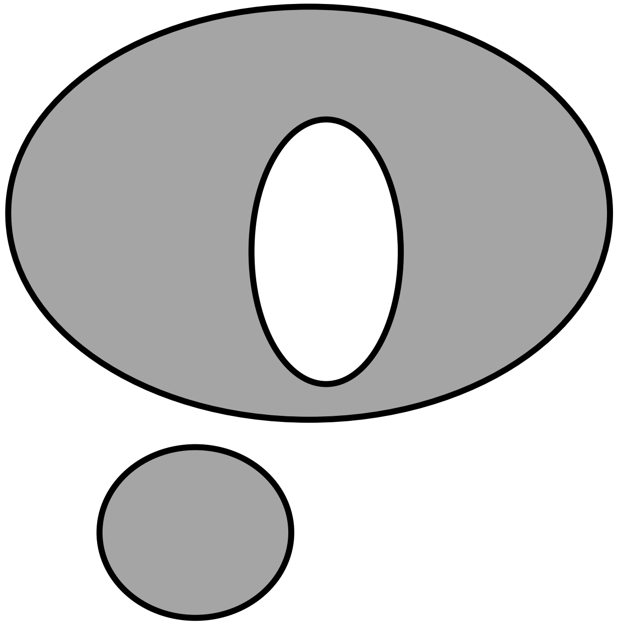
- Technical contributions
  - Extension of the single-shape method of [Zeng 20]
  - Formulation as a discrete optimization problem
  - An efficient and effective solver

Topological Operators

Washington University in St. Louis
James McKelvey School of Engineering

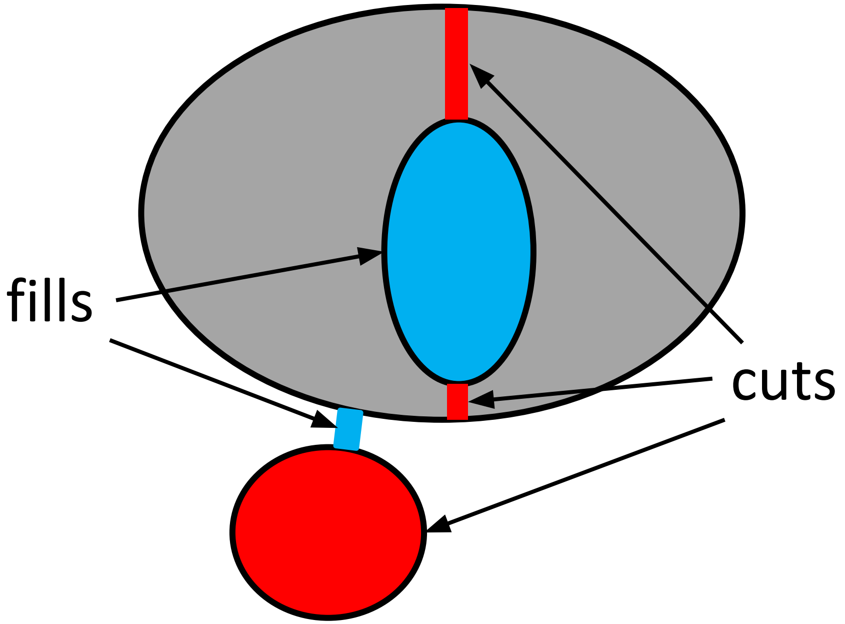


Single-shape Simplification [Zeng 20]



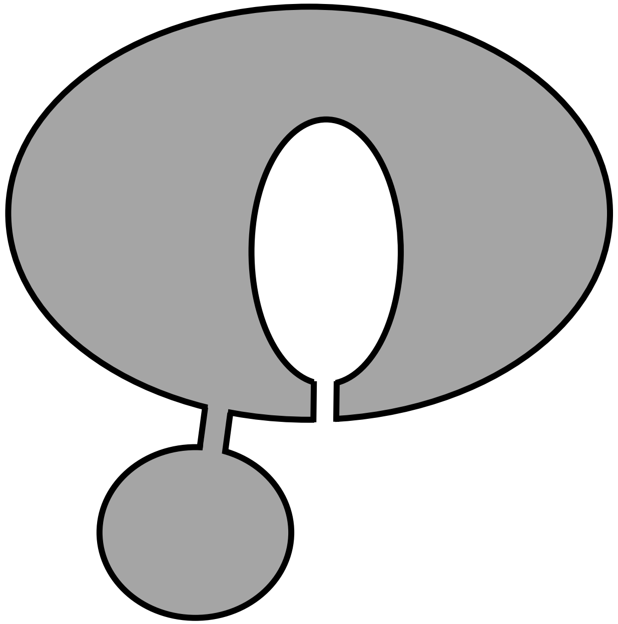
Single-shape Simplification [Zeng 20]

- Compute candidate cuts and fills
  - Applying a cut or fill removes one or more features
  - Each candidate associated with a cost
- Select a subset of candidates that:
  - Maximally removes topological features
  - Minimizes total cost



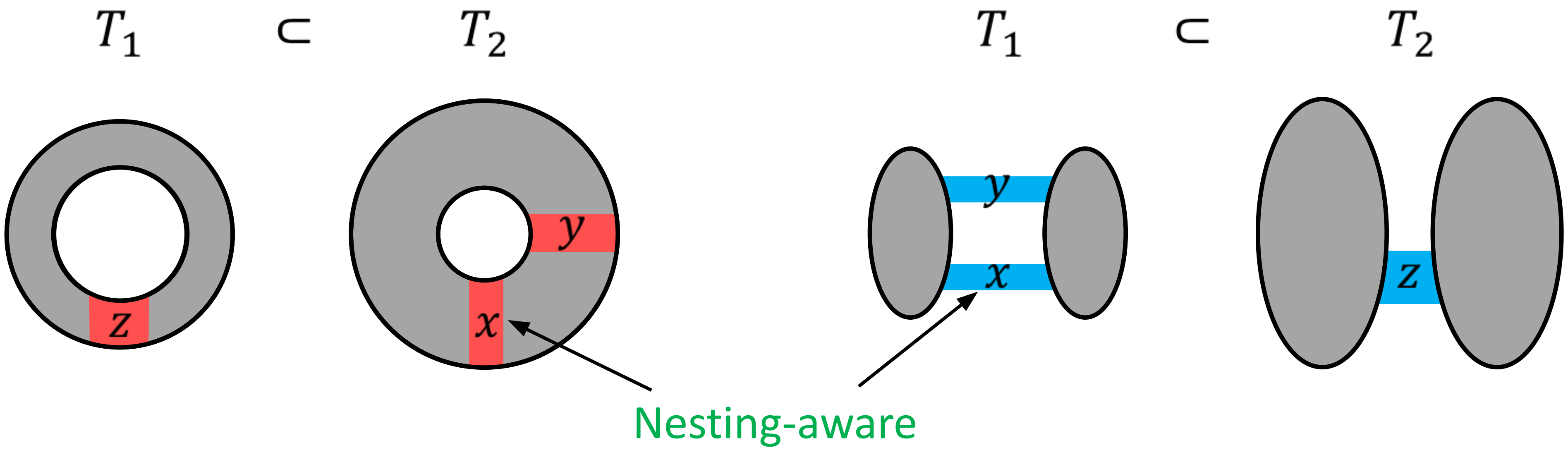
Single-shape Simplification [Zeng 20]

- Compute candidate cuts and fills
  - Applying a cut or fill removes one or more features
  - Each candidate associated with a cost
- Select a subset of candidates that:
  - Maximally removes topological features
  - Minimizes total cost
- Solved as a graph labelling problem



Nesting-Aware Candidates

Washington University in St. Louis
James McKelvey School of Engineering



- Cut x may be used (if z is also used)
- Cut y may never be used

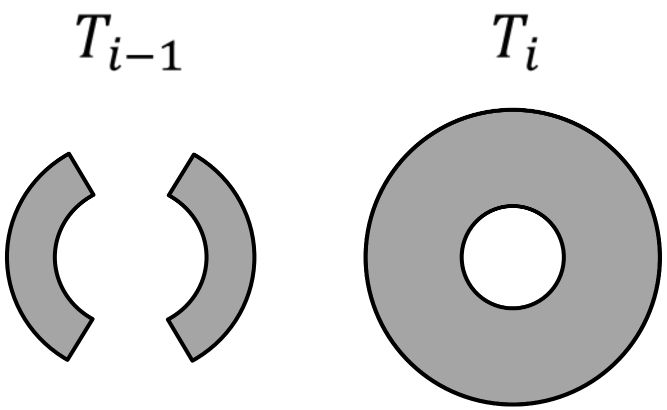
- Fill x may be used (if z is also used)
- Fill y may never be used

Optimization Problem

Washington University in St. Louis
James McKelvey School of Engineering

- Given:
  - Nested shapes $\{T_1 \subset \cdots \subset T_n\}$
  - Nesting-aware candidates $\{X_1, \ldots, X_n\}$, each with a cost
- Label candidates as inside (1) or outside (0) to:
  - Maximally remove topological features of each shape
  - Minimize total costs of 0-labelled cuts and 1-labelled fills
  - Maintain nesting

[Zeng 20]

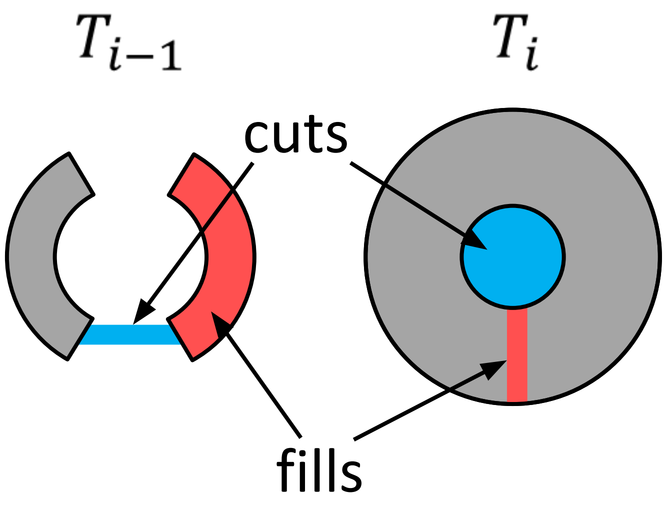


Optimization Problem

Washington University in St. Louis
James McKelvey School of Engineering

- Given:
  - Nested shapes $\{T_1 \subset \cdots \subset T_n\}$
  - Nesting-aware candidates $\{X_1, \dots, X_n\}$, each with a cost
- Label candidates as inside (1) or outside (0) to:
  - Maximally remove topological features of each shape
  - Minimize total costs of 0-labelled cuts and 1-labelled fills
  - Maintain nesting

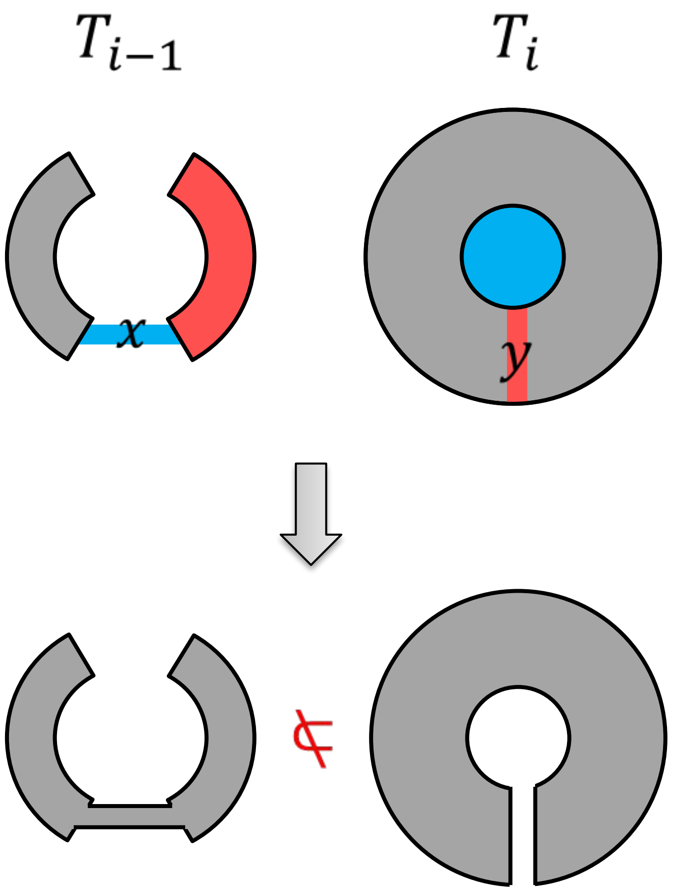
[Zeng 20]



Optimization Problem

- Given:
  - Nested shapes $\{T_1 \subset \cdots \subset T_n\}$
  - Nesting-aware candidates $\{X_1, \ldots, X_n\}$, each with a cost
- Label candidates as inside (1) or outside (0) to:
  - Maximally remove topological features of each shape
  - Minimize total costs of 0-labelled cuts and 1-labelled fills

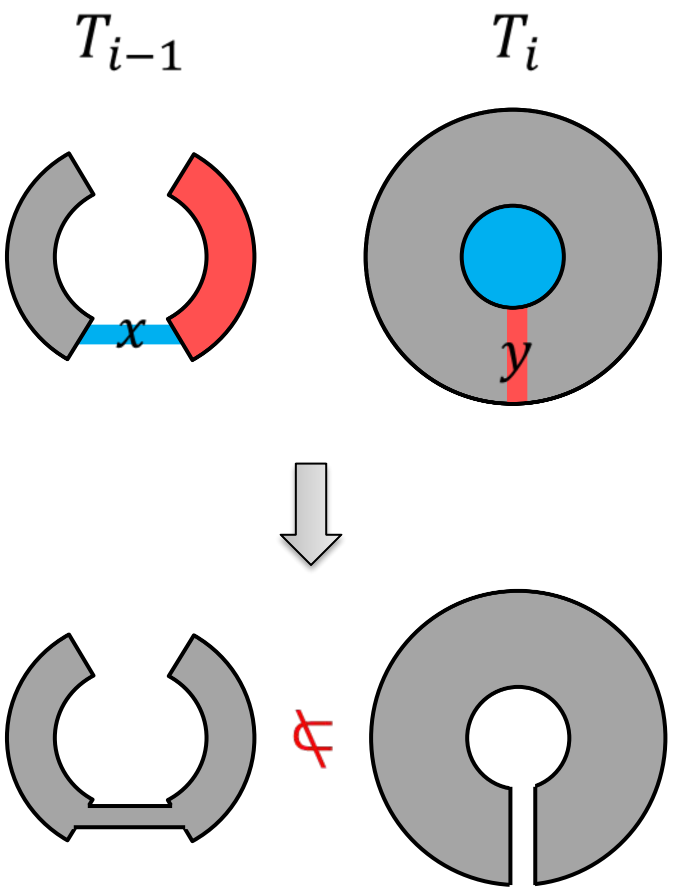
  [Zeng 20]
  - Maintain nesting



Optimization Problem

- Given:
  - Nested shapes $\{T_1 \subset \cdots \subset T_n\}$
  - Nesting-aware candidates $\{X_1, \ldots, X_n\}$, each with a cost
- Label candidates as inside (1) or outside (0) to:
  - Maximally remove topological features of each shape
  - Minimize total costs of 0-labelled cuts and 1-labelled fills

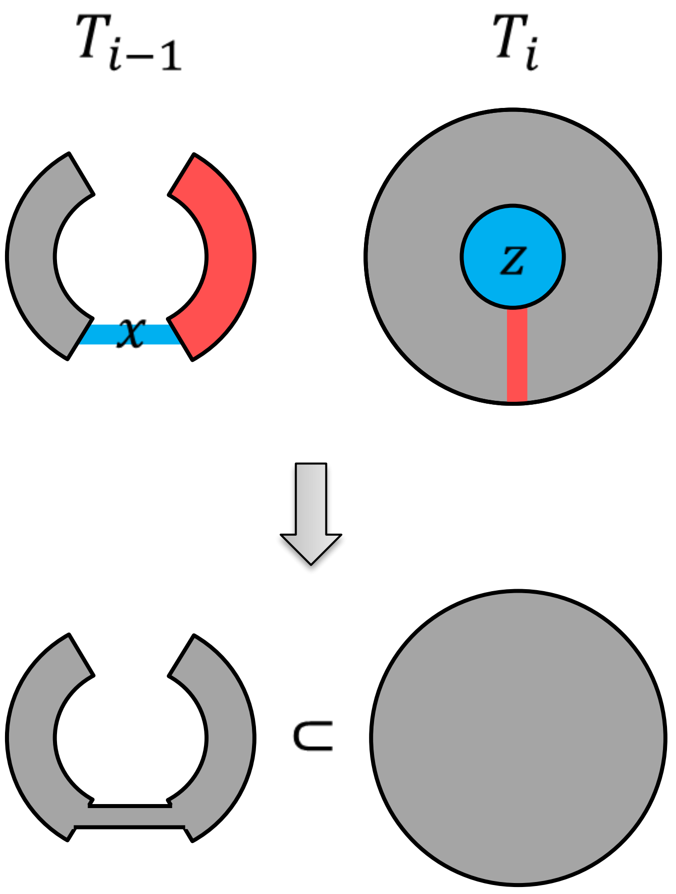
  [Zeng 20]
  - Avoid conflicting labels
    - Conflict: $x \in X_{i-1}$ overlaps with $y \in X_i$, x has label 1, y has label 0



Optimization Problem

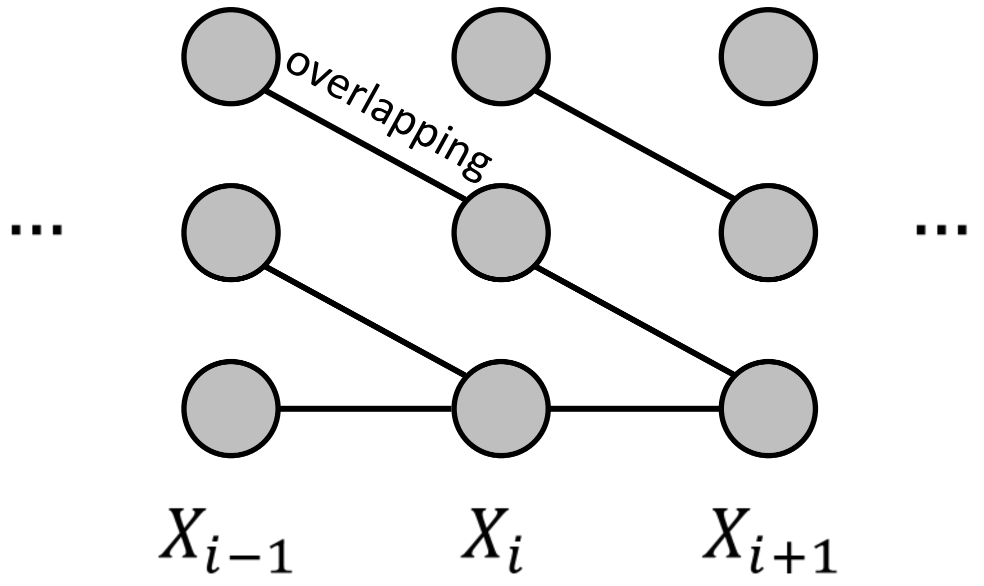
- Given:
  - Nested shapes $\{T_1 \subset \cdots \subset T_n\}$
  - Nesting-aware candidates $\{X_1, \ldots, X_n\}$, each with a cost
- Label candidates as inside (1) or outside (0) to:
  - Maximally remove topological features of each shape
  - Minimize total costs of 0-labelled cuts and 1-labelled fills
  - Avoid conflicting labels
    - Conflict: $x \in X_{i-1}$ overlaps with $y \in X_i$, x has label 1, y has label 0

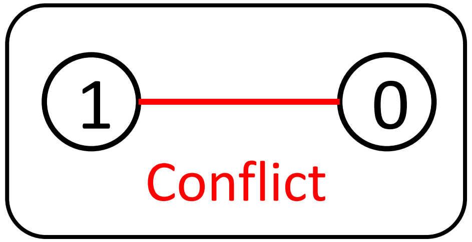
[Zeng 20]



Solver 1: Label Propagation

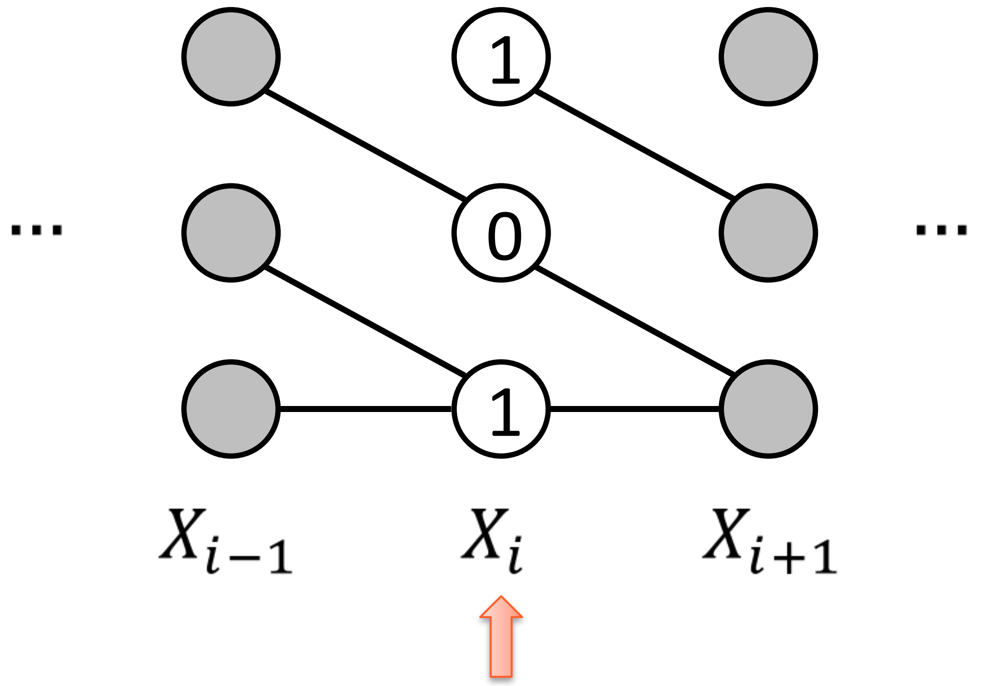
- Propagate labels from one shape to others while avoiding conflicts
  - Use [Zeng 20] to optimize labels on each shape





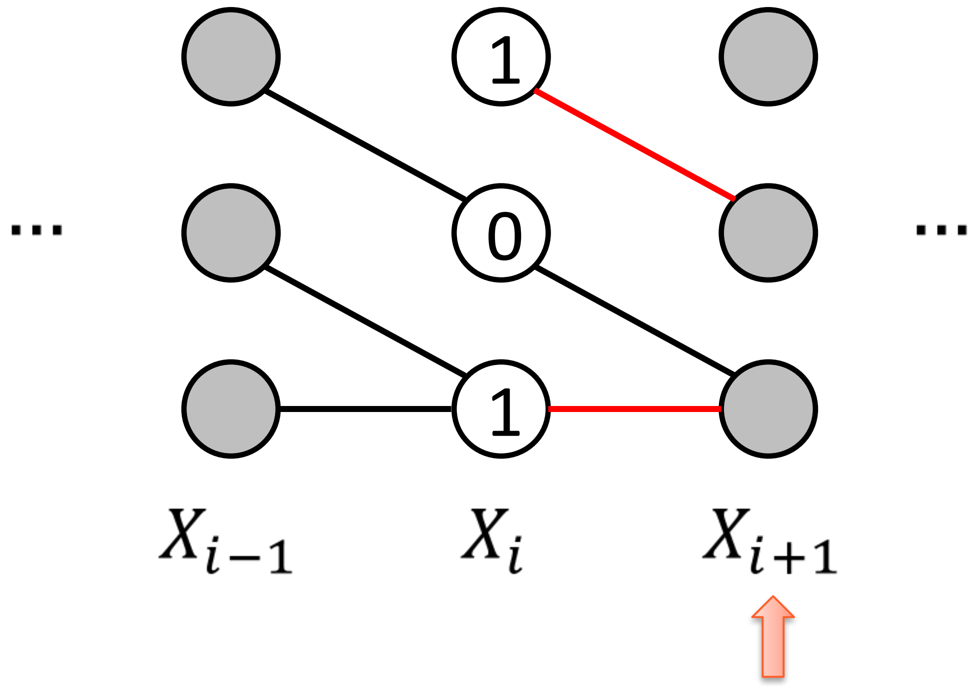
Solver 1: Label Propagation

- Propagate labels from one shape to others while avoiding conflicts
  - Use [Zeng 20] to optimize labels on each shape



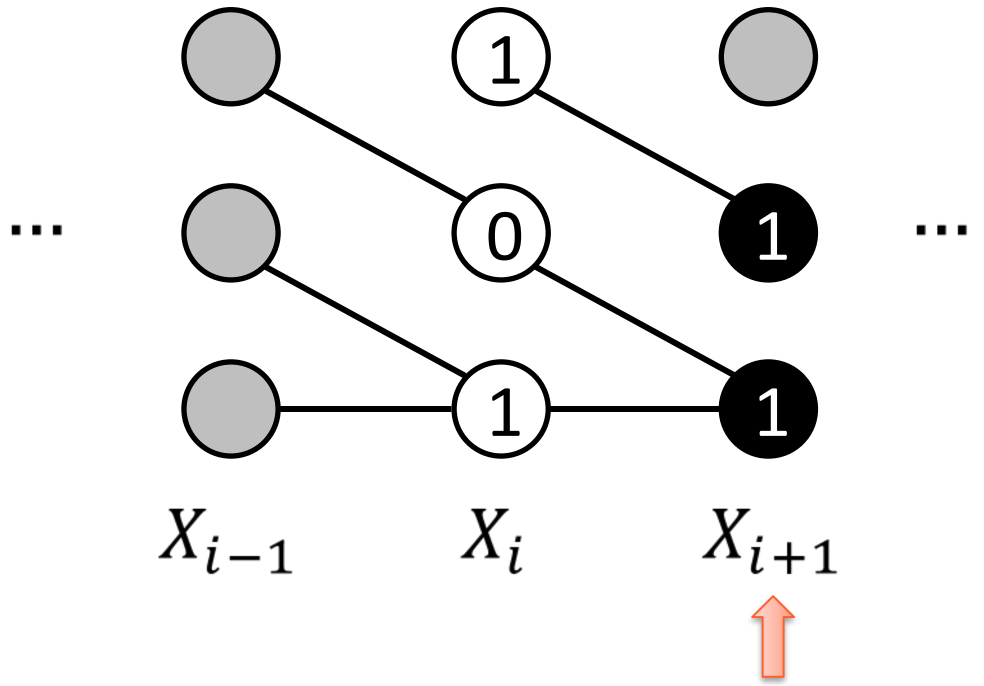
Solver 1: Label Propagation

- Propagate labels from one shape to others while avoiding conflicts
  - Use [Zeng 20] to optimize labels on each shape



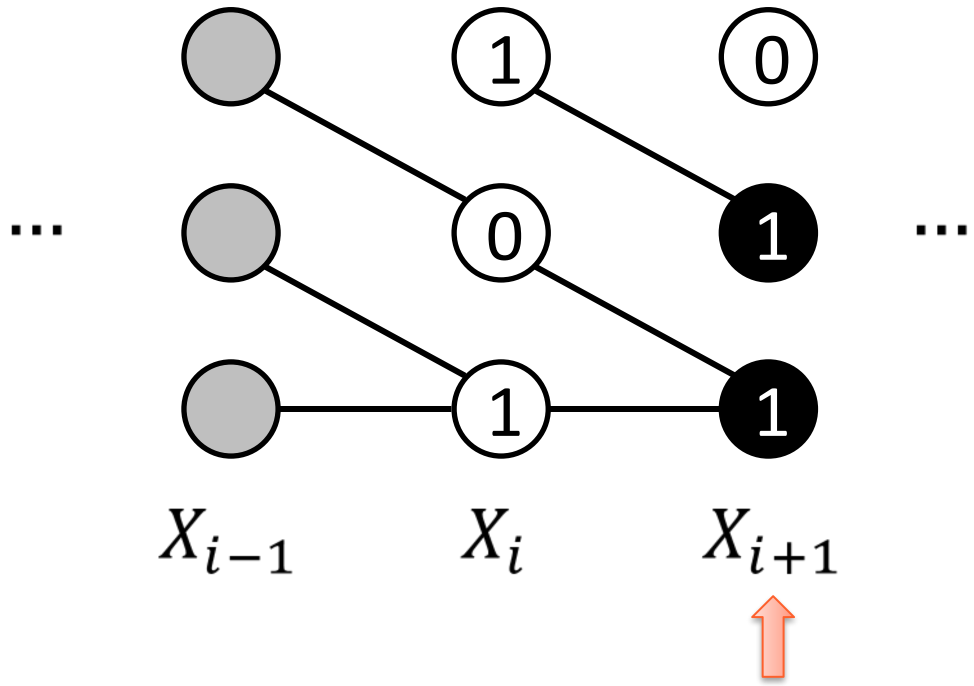
Solver 1: Label Propagation

- Propagate labels from one shape to others while avoiding conflicts
  - Use [Zeng 20] to optimize labels on each shape



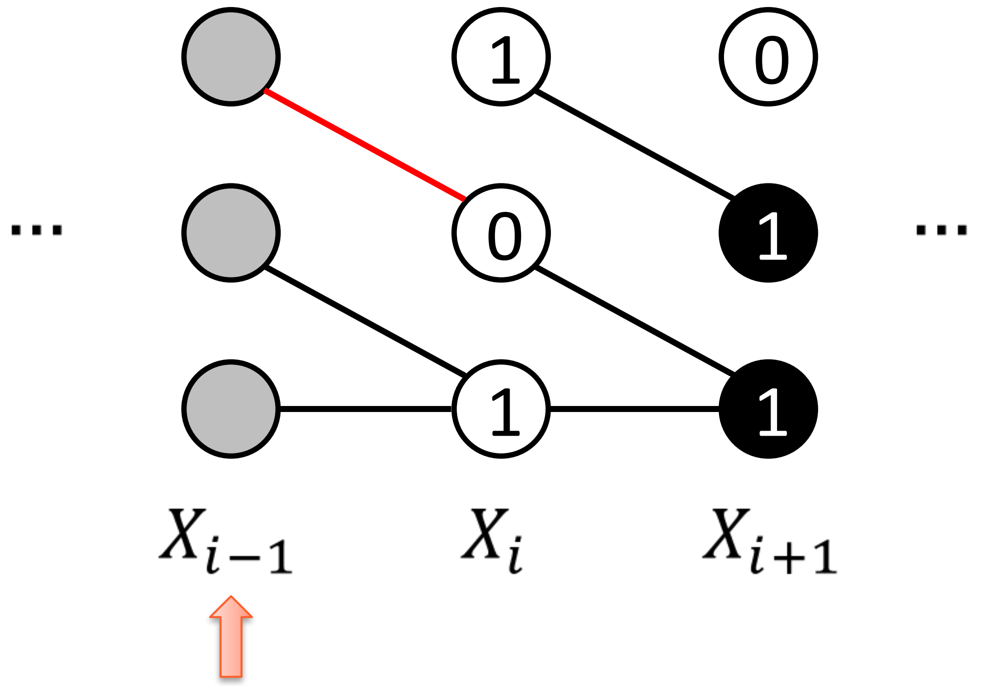
Solver 1: Label Propagation

- Propagate labels from one shape to others while avoiding conflicts
  - Use [Zeng 20] to optimize labels on each shape



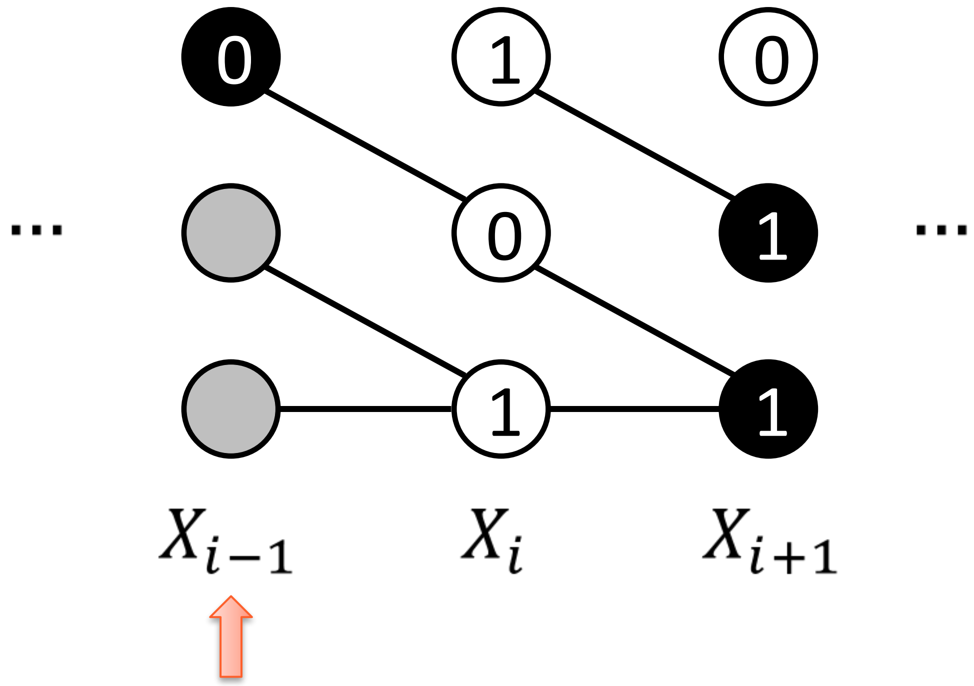
Solver 1: Label Propagation

- Propagate labels from one shape to others while avoiding conflicts
  - Use [Zeng 20] to optimize labels on each shape



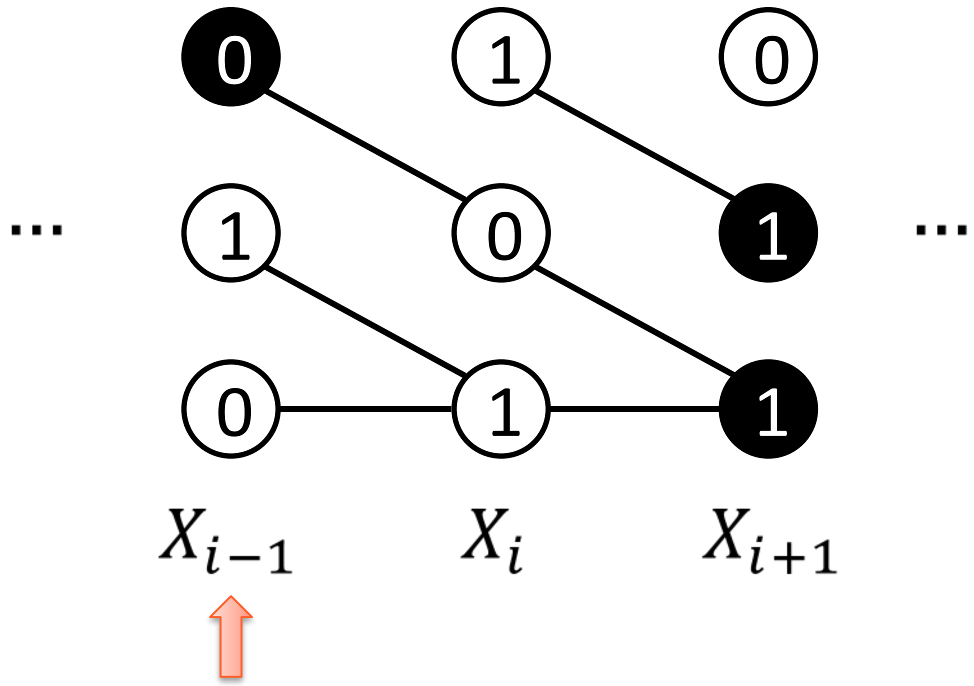
Solver 1: Label Propagation

- Propagate labels from one shape to others while avoiding conflicts
  - Use [Zeng 20] to optimize labels on each shape



Solver 1: Label Propagation

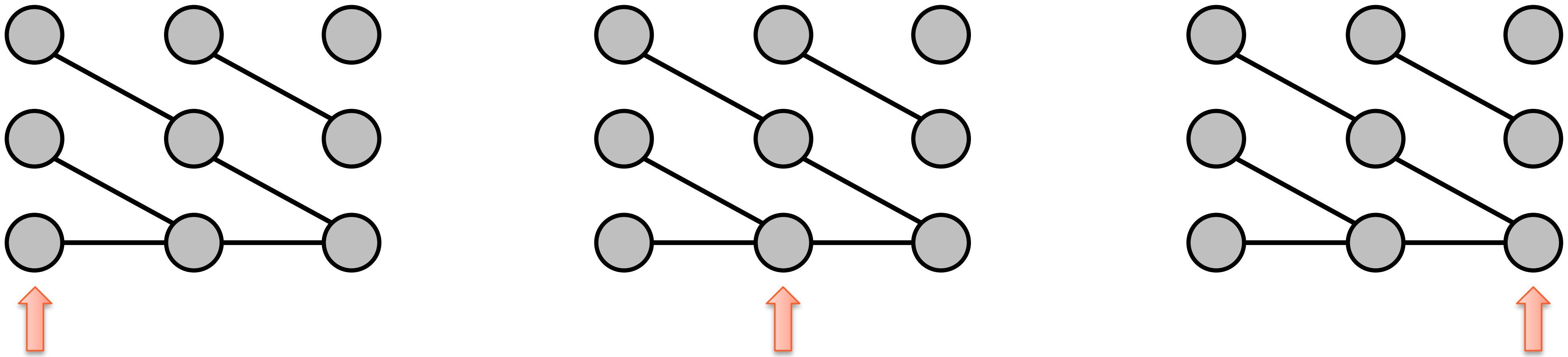
- Propagate labels from one shape to others while avoiding conflicts
  - Use [Zeng 20] to optimize labels on each shape



Solver 1: Label Propagation

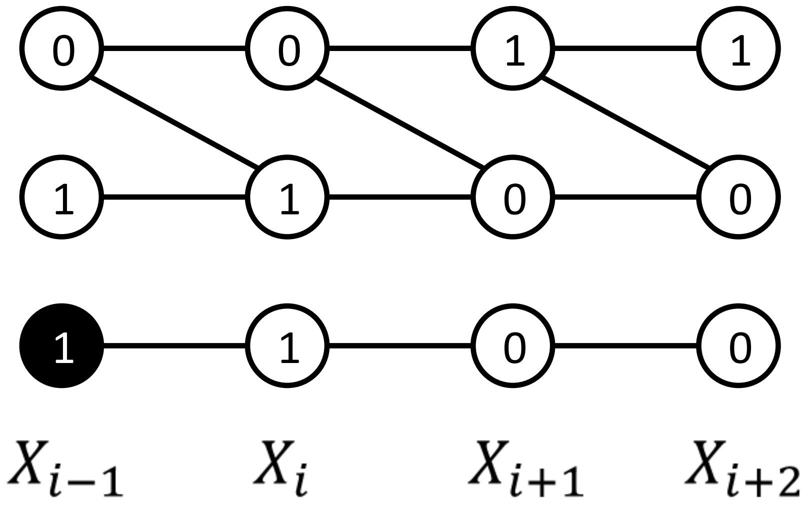
Washington University in St. Louis
James McKelvey School of Engineering

- Propagate labels from one shape to others while avoiding conflicts
  - Among all starting shapes, take the solution with the minimal topology and costs
  - Guarantees to be free of conflicts
  - May not be optimal in topology simplicity or geometric cost



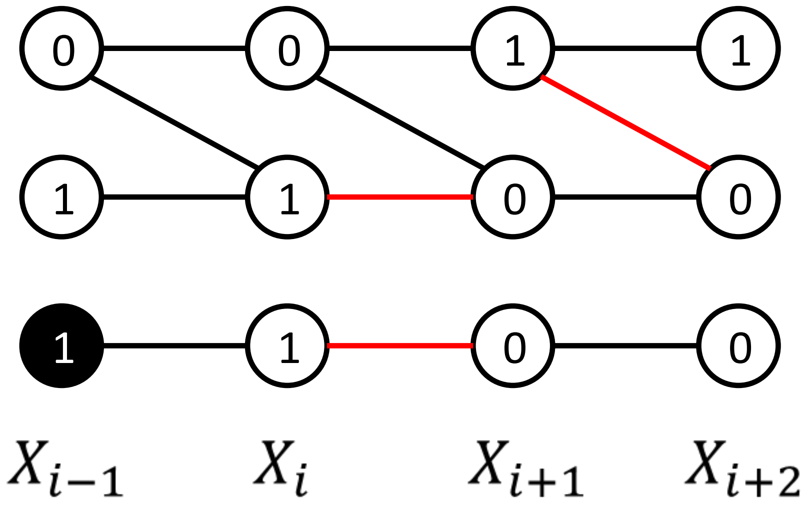
Solver 2: State-space Search

- *State*: a labelling of all candidates, and a set of constrained candidates
  - In a queue sorted by topology + geometric cost



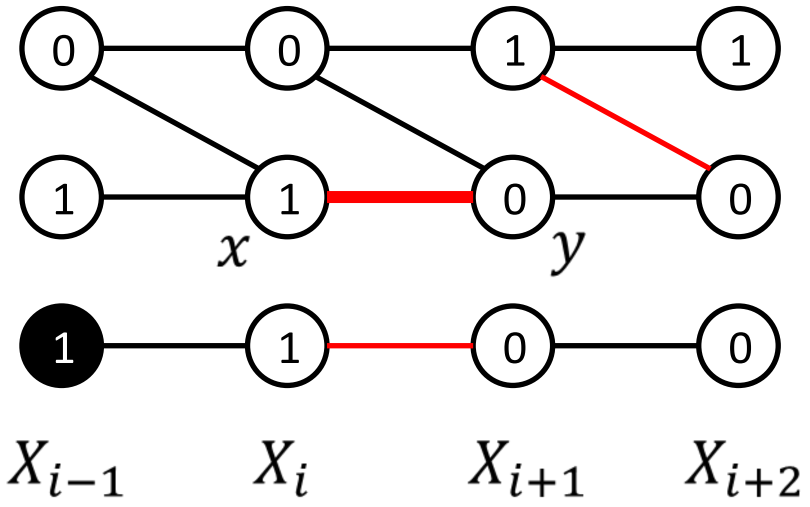
Solver 2: State-space Search

- *State*: a labelling of all candidates, and a set of constrained candidates
  - In a queue sorted by topology + geometric cost
- If the popped state has conflicts:



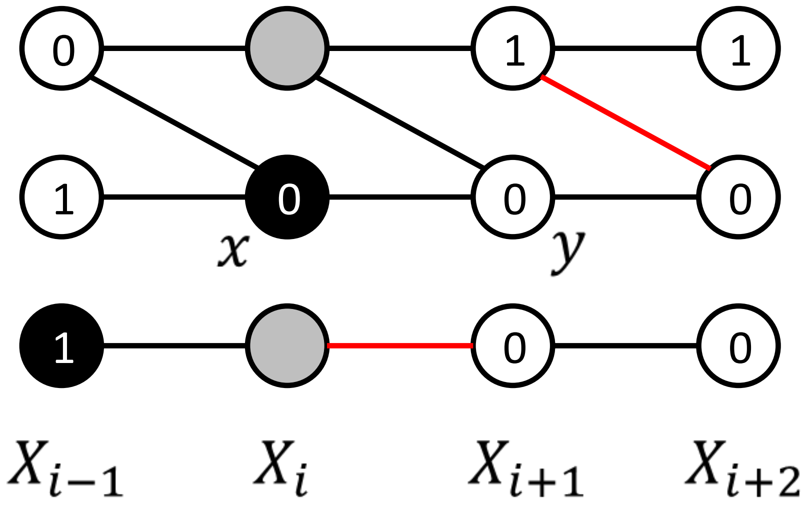
Solver 2: State-space Search

- *State*: a labelling of all candidates, and a set of constrained candidates
  - In a queue sorted by topology + geometric cost
- If the popped state has conflicts:
  - Pick a conflict $\{x \in X_i, y \in X_{i+1}\}$
  - Create 2 new states by either constraining x's label to be 0 or y's label to be 1



Solver 2: State-space Search

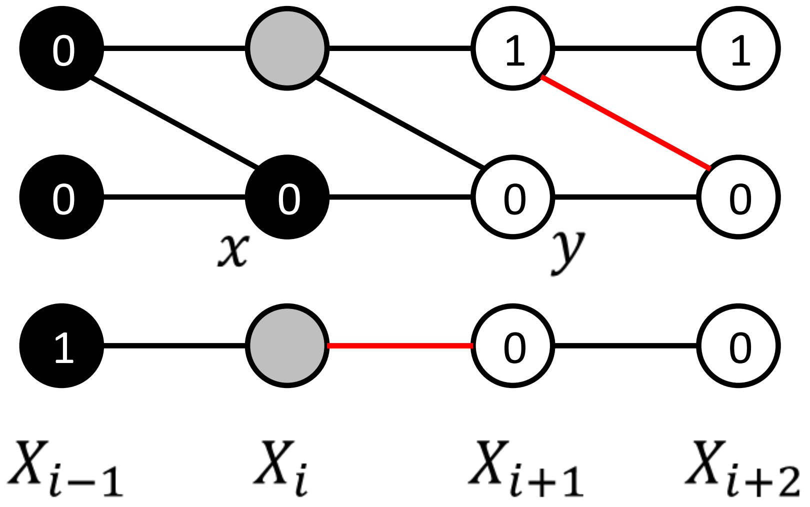
- *State*: a labelling of all candidates, and a set of constrained candidates
  - In a queue sorted by topology + geometric cost
- If the popped state has conflicts:
  - Pick a conflict $\{x \in X_i, y \in X_{i+1}\}$
  - Create 2 new states by either constraining x's label to be 0 or y's label to be 1



Solver 2: State-space Search

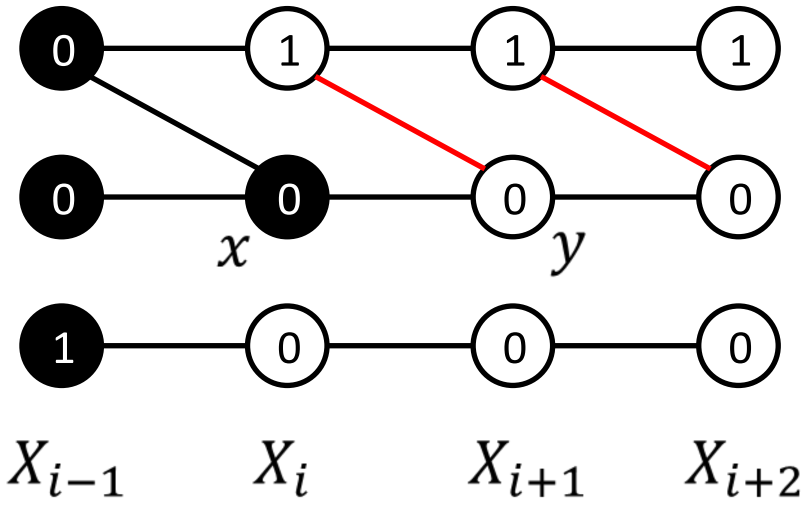
Washington University in St. Louis
James McKelvey School of Engineering

- *State*: a labelling of all candidates, and a set of constrained candidates
  - In a queue sorted by topology + geometric cost
- If the popped state has conflicts:
  - Pick a conflict $\{x \in X_i, y \in X_{i+1}\}$
  - Create 2 new states by either constraining x's label to be 0 or y's label to be 1



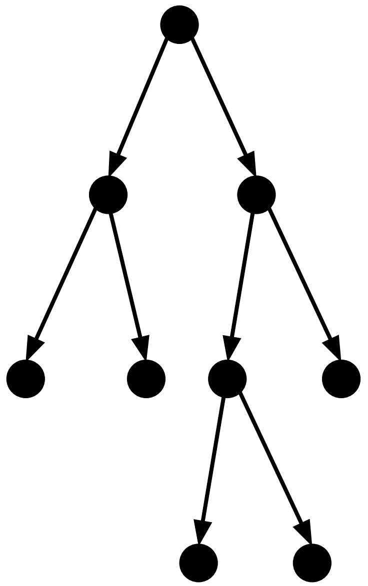
Solver 2: State-space Search

- *State*: a labelling of all candidates, and a set of constrained candidates
  - In a queue sorted by topology + geometric cost
- If the popped state has conflicts:
  - Pick a conflict $\{x \in X_i, y \in X_{i+1}\}$
  - Create 2 new states by either constraining x's label to be 0 or y's label to be 1
- Terminate otherwise



Solver 2: State-space Search

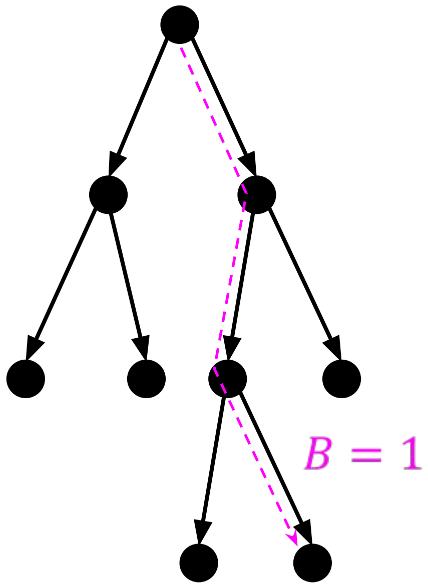
- Best-first search in a binary tree of states
- Returns optimal conflict-free labelling
  - Assuming [Zeng 20] is optimal
- High computational cost
  - # iterations can be exponential in total # candidates



Solver 3: Beam Search

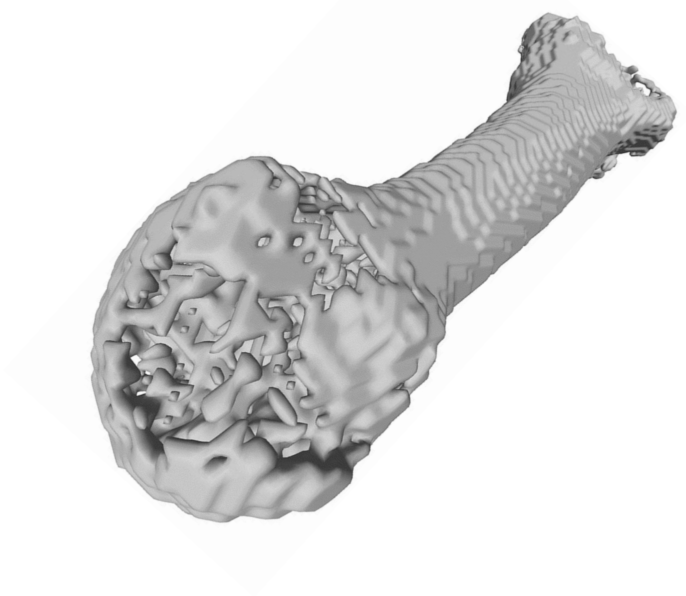
Washington University in St. Louis
James McKelvey School of Engineering

- Limit queue size to a constant B
  - Keep only best B states
- Trade off optimality for efficiency
  - # iterations linear in total # candidates



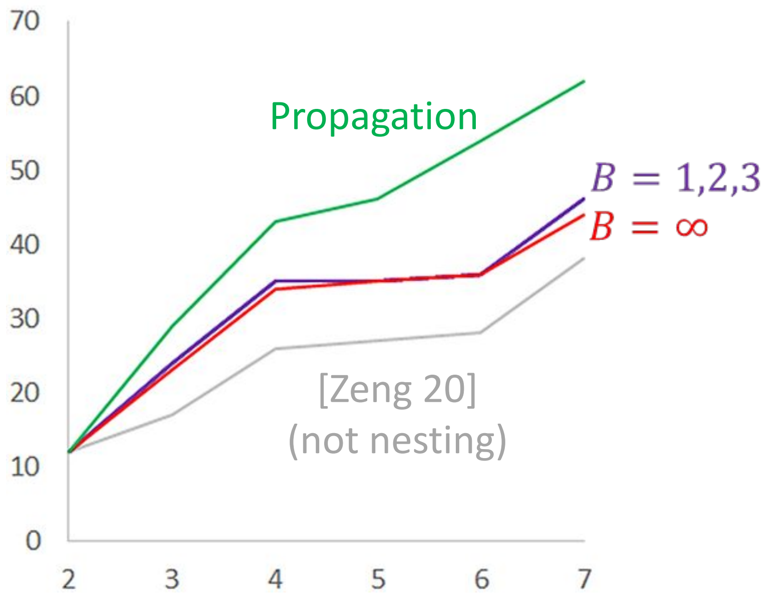
Solver Comparison

Level set of
CT bone scan



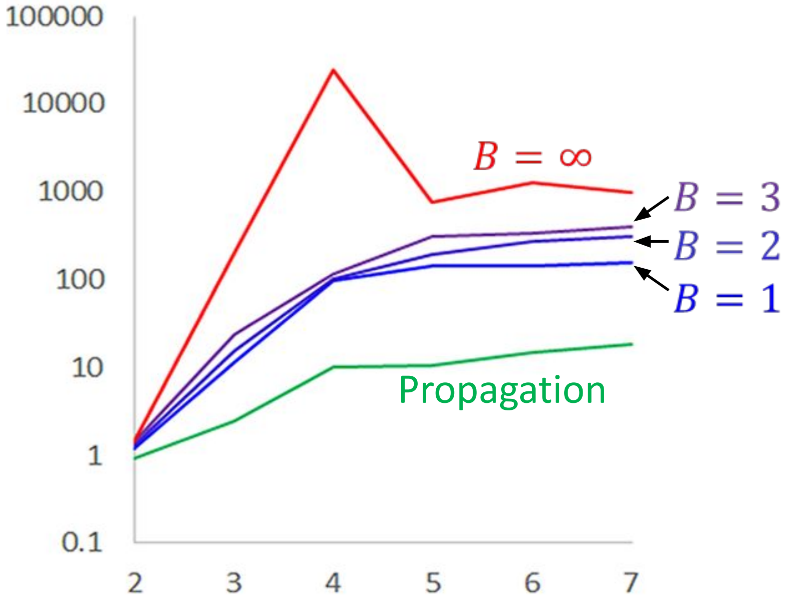
components: 31
handles: 371
voids: 106

Topo. features



Level sets

Running time (secs)



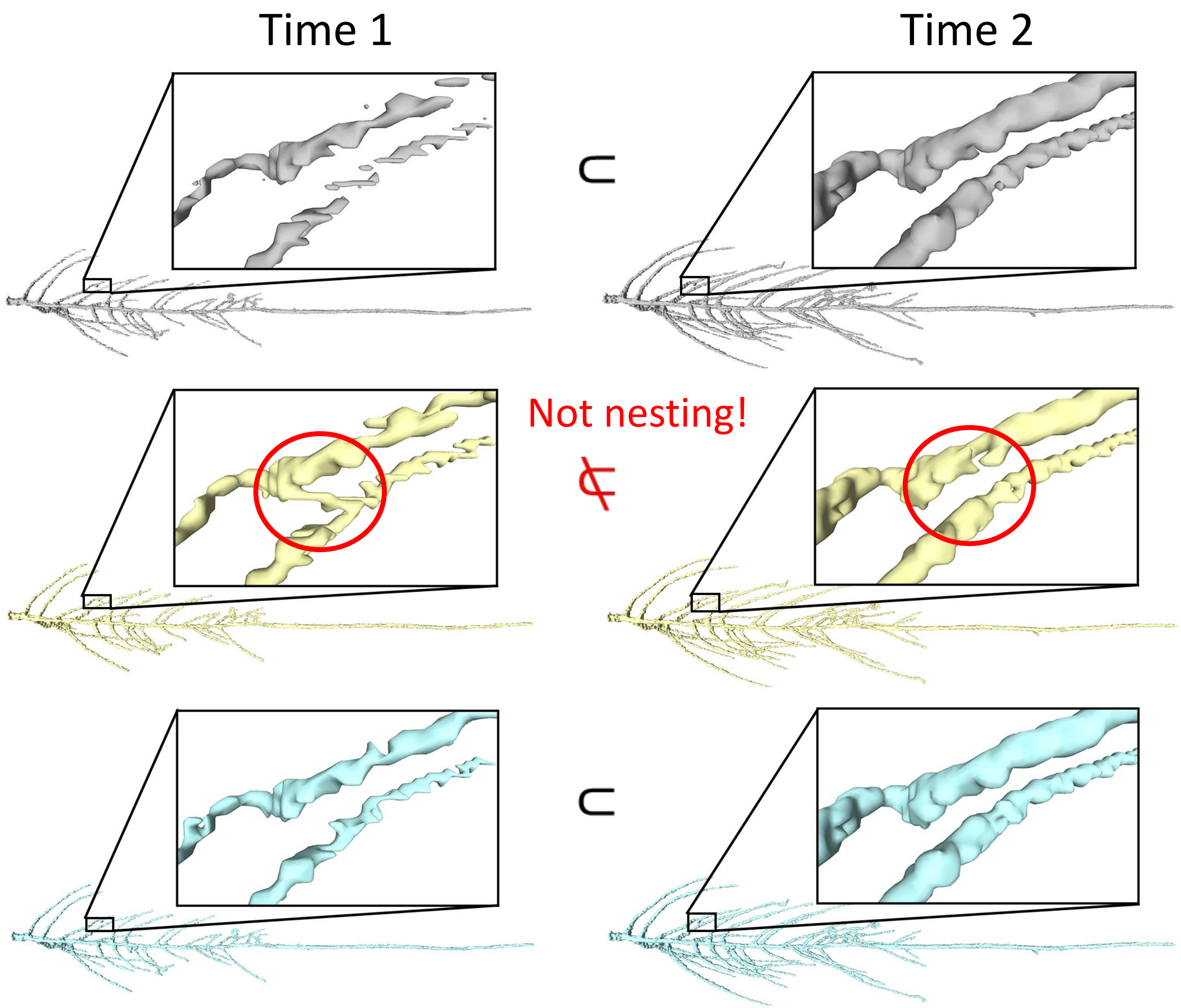
Level sets

Results: Roots

Input:

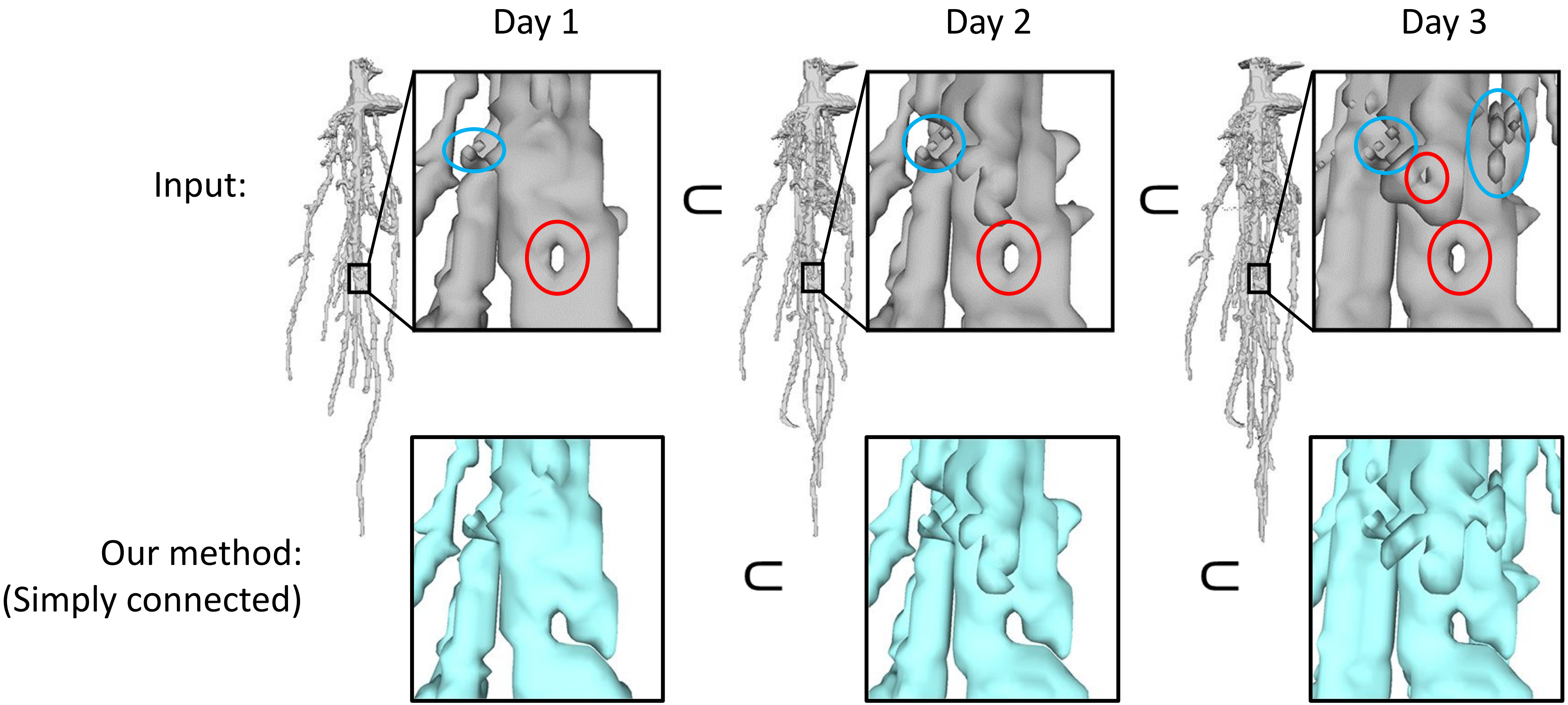
Single-shape [Zeng 20]:
(Simply connected)

Our method:
(Simply connected)



Results: Roots

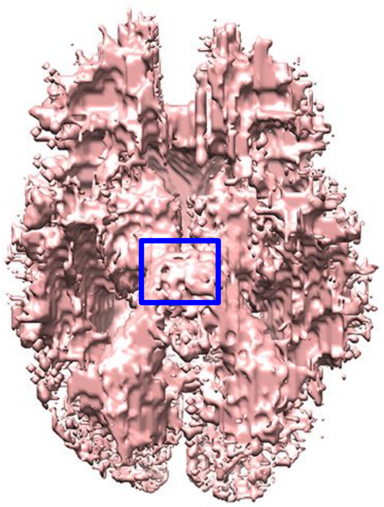
Washington University in St. Louis
James McKelvey School of Engineering



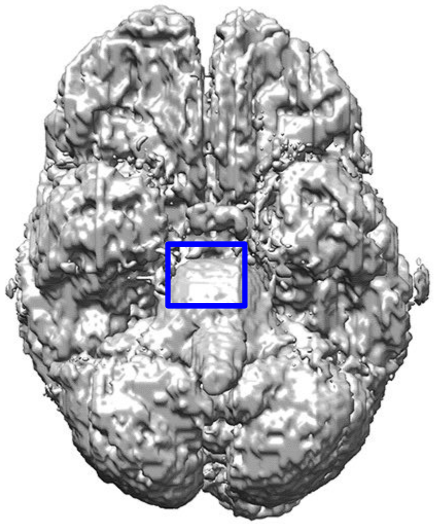
Results: Brain

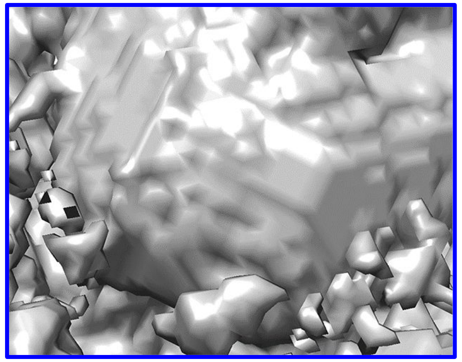
Washington University in St. Louis
James McKelvey School of Engineering

Cerebrospinal fluid

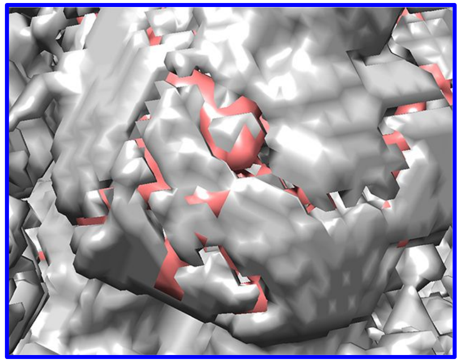


White matter

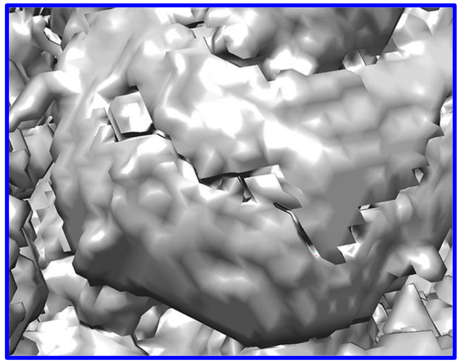




Input (nested)



Single-shape [Zeng 20]
(simply connected; not nested)

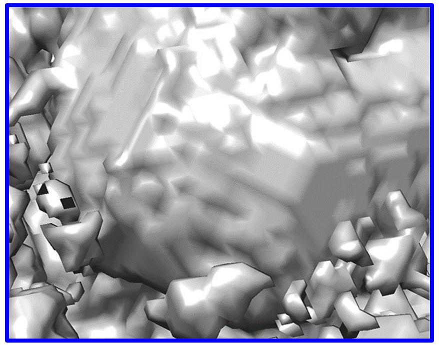


Our method
(simply connected and nested)

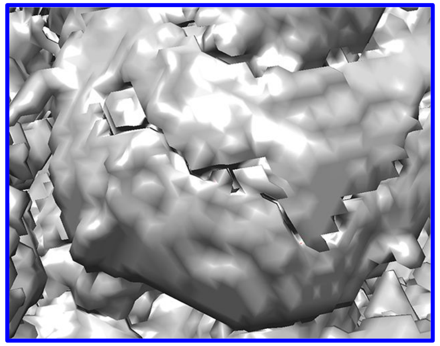
Limitations and Future Works

Washington University in St. Louis
James McKelvey School of Engineering

- Need more "natural-looking" candidates and "semantic" geometric costs
- Handling non-cubical complexes
- How to simplify a (not necessarily nesting) shape collection in a consistent way?



Input



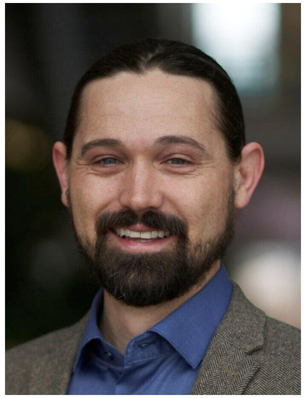
Our method

Acknowledgement

Washington University in St. Louis
James McKelvey School of Engineering

- Danforth Plant Science Center
  - Chris Topp, Mao Li
- Funding
  - NSF ABI-1759836, NSF EF-1921728, AF-1907612 and AF-2106672
  - WashU Imaging Science Pathway Fellowship (Dan)

Danforth Plant Science Center



Chris Topp

Dan Zeng