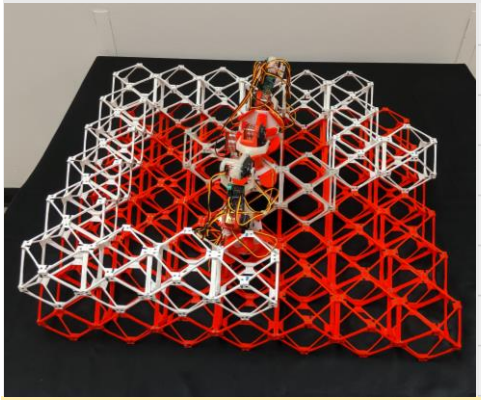
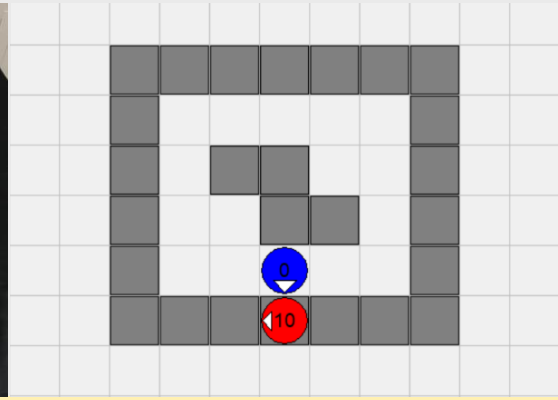
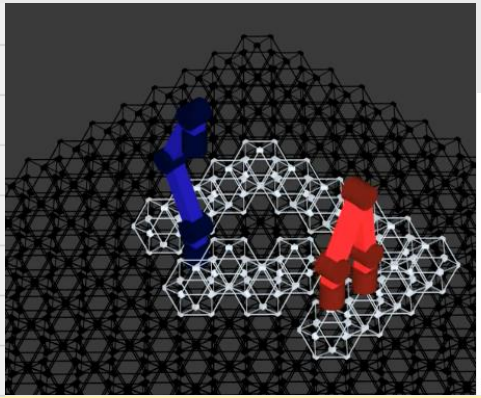
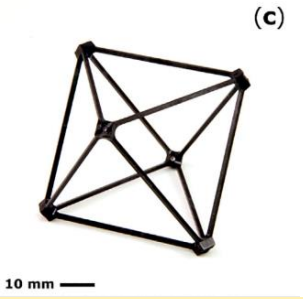
Technische Universität Braunschweig





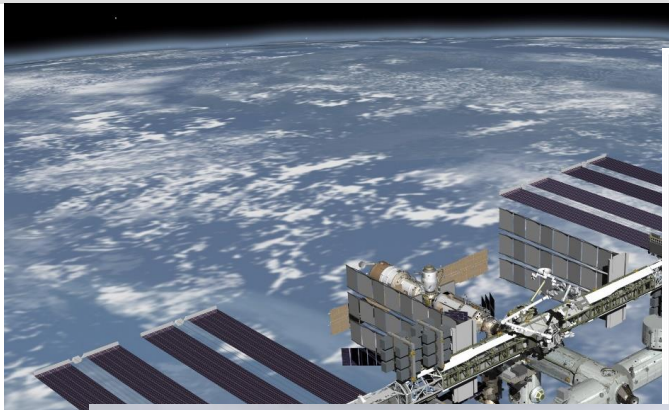


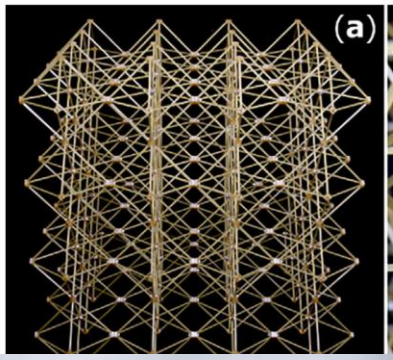


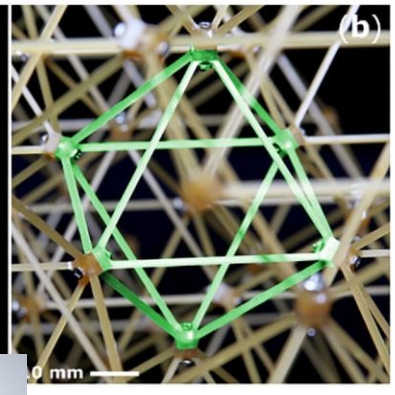
Recognition and Reconfiguration of Lattice-Based Cellular Structures by Simple Robots

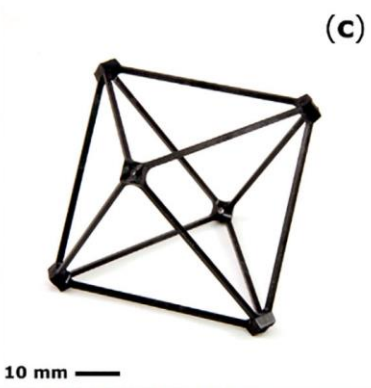
Amira Abdel-Rahman, Aaron T. Becker, Daniel E. Biediger, Kenneth C. Cheung, Sándor P. Fekete, Benjamin Jenett, **Eike Niehs**, Christian Scheffer, Arne Schmidt, and Mike Yannuzzi

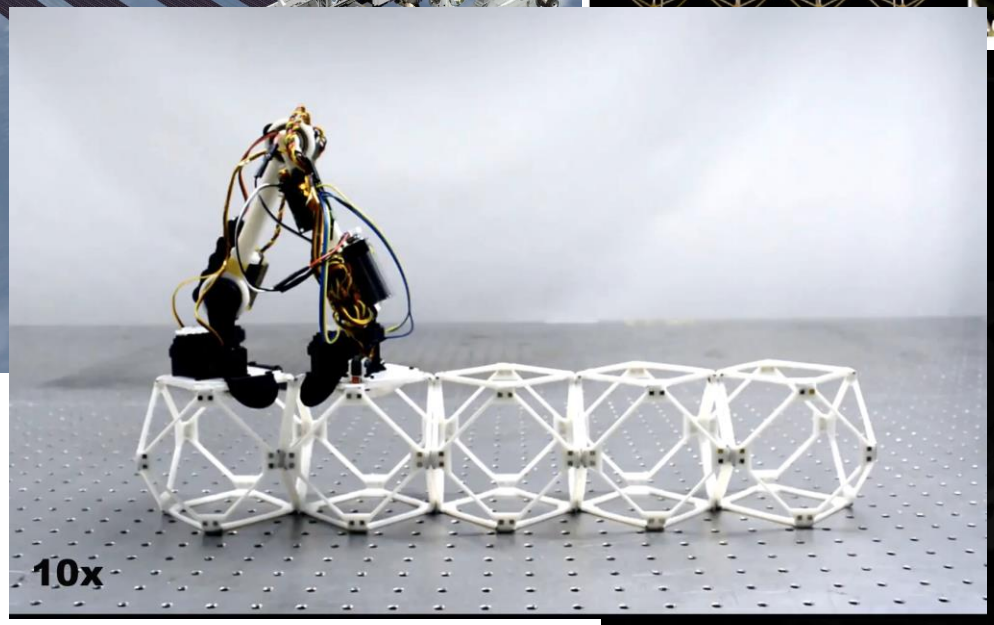
Motivation

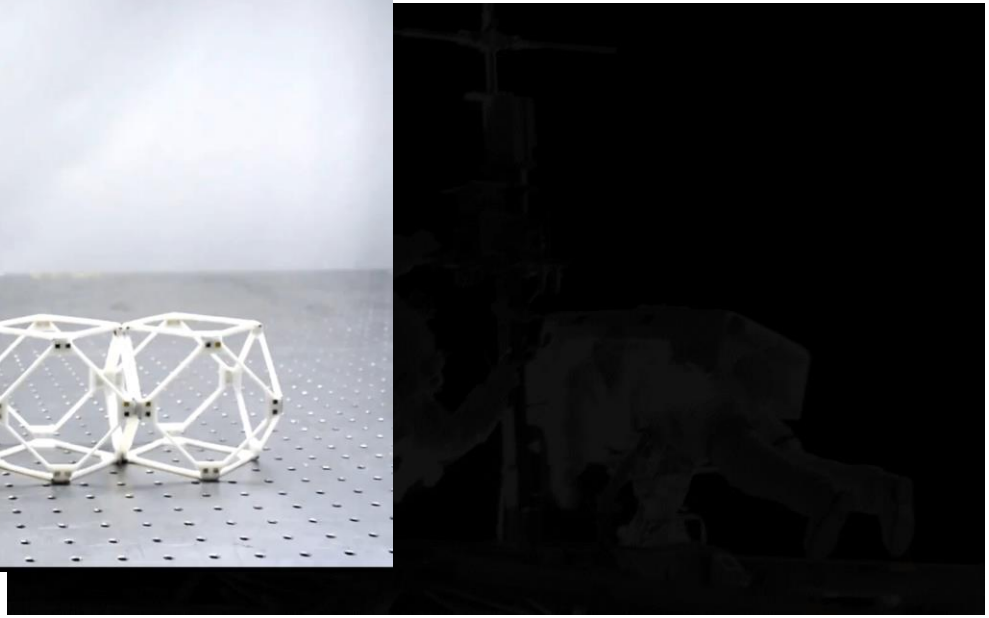




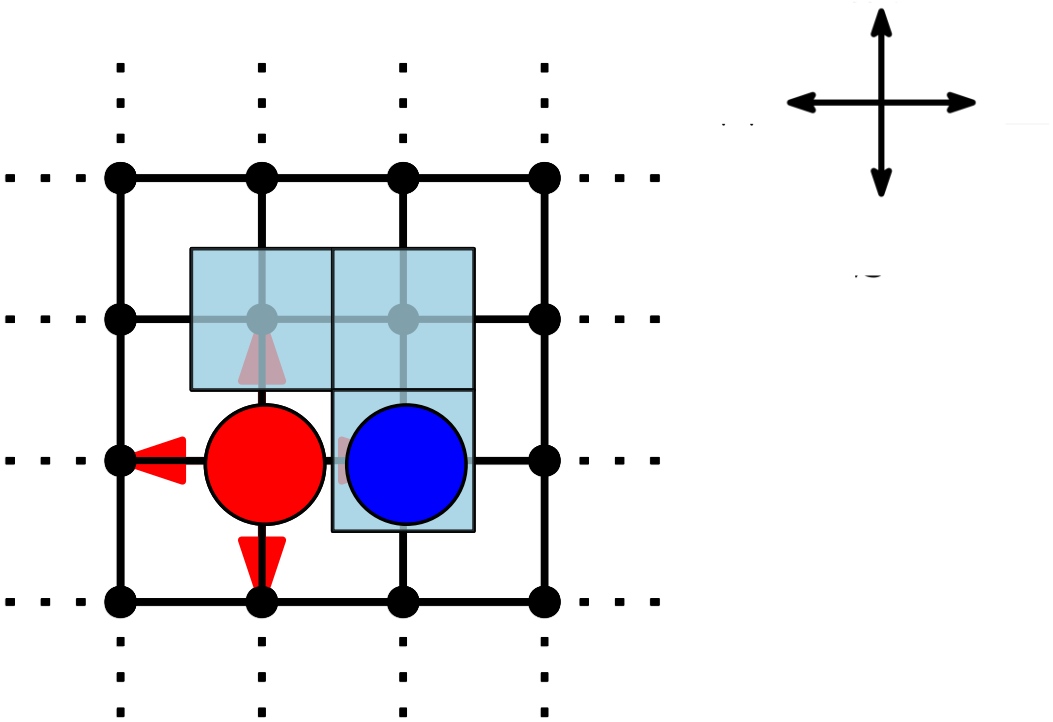








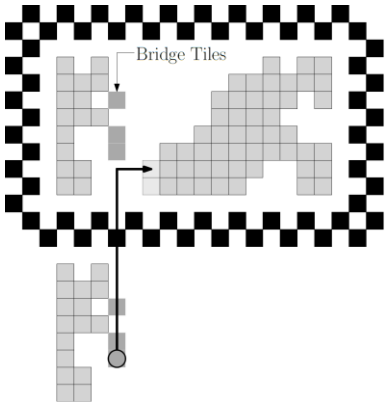
Robots-on-Tiles Model



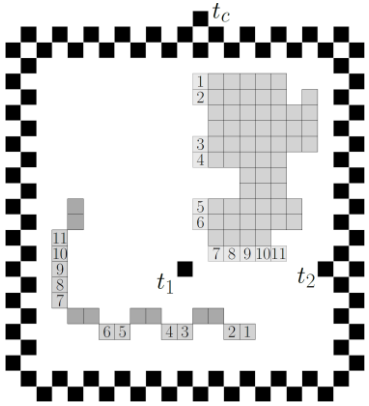
CADbots: Algorithmic Aspects of Manipulating Programmable Matter with Finite Automata

Sándor P. Fekete[1], Robert Gmyr[2], Sabrina Hugo[1], Phillip Keldenich[1], Christian Scheffer[1], and Arne Schmidt[1]

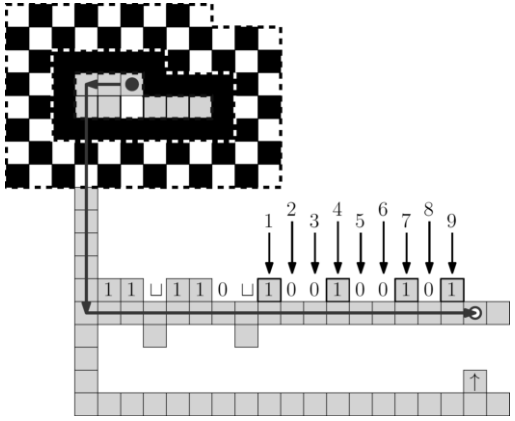
[1]Department of Computer Science, TU Braunschweig, Germany.
{fekete, hugo, keldenich, scheffer, aschmidt}@ibr.cs.tu-bs.de
[2]Department of Computer Science, University of Houston, USA rgmyr@uh.edu



Copying

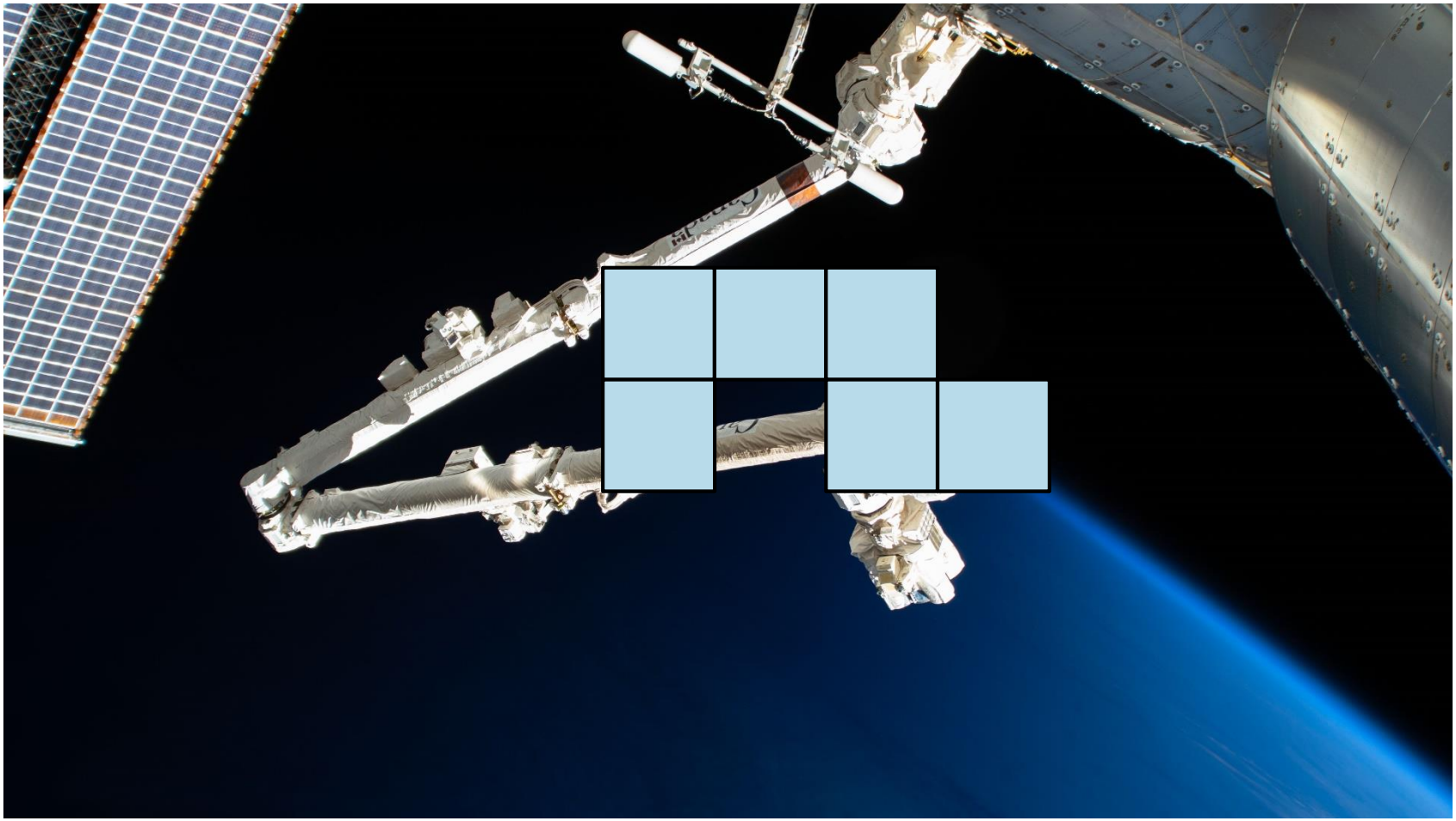


Rotating

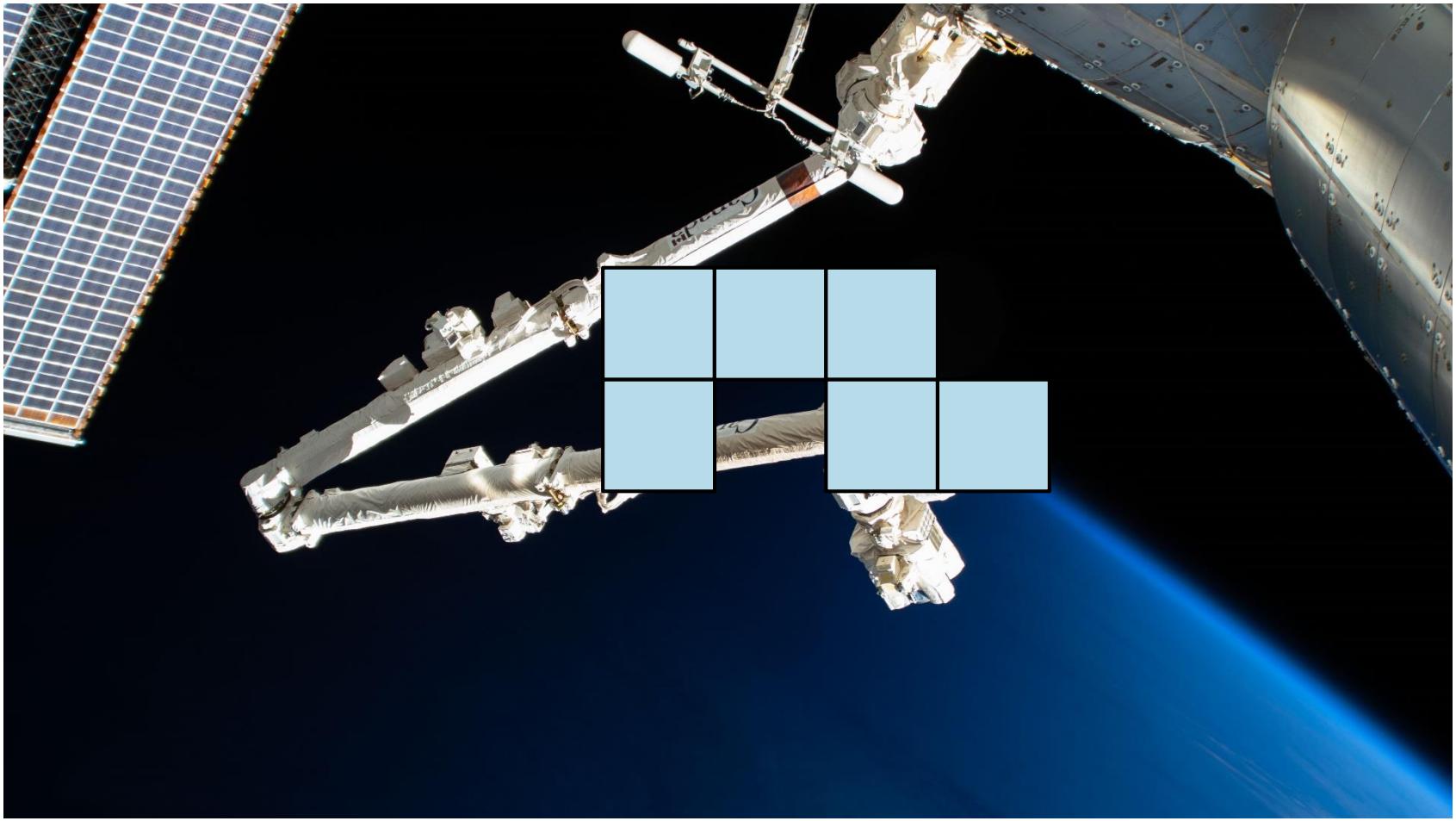


TM-Transformation

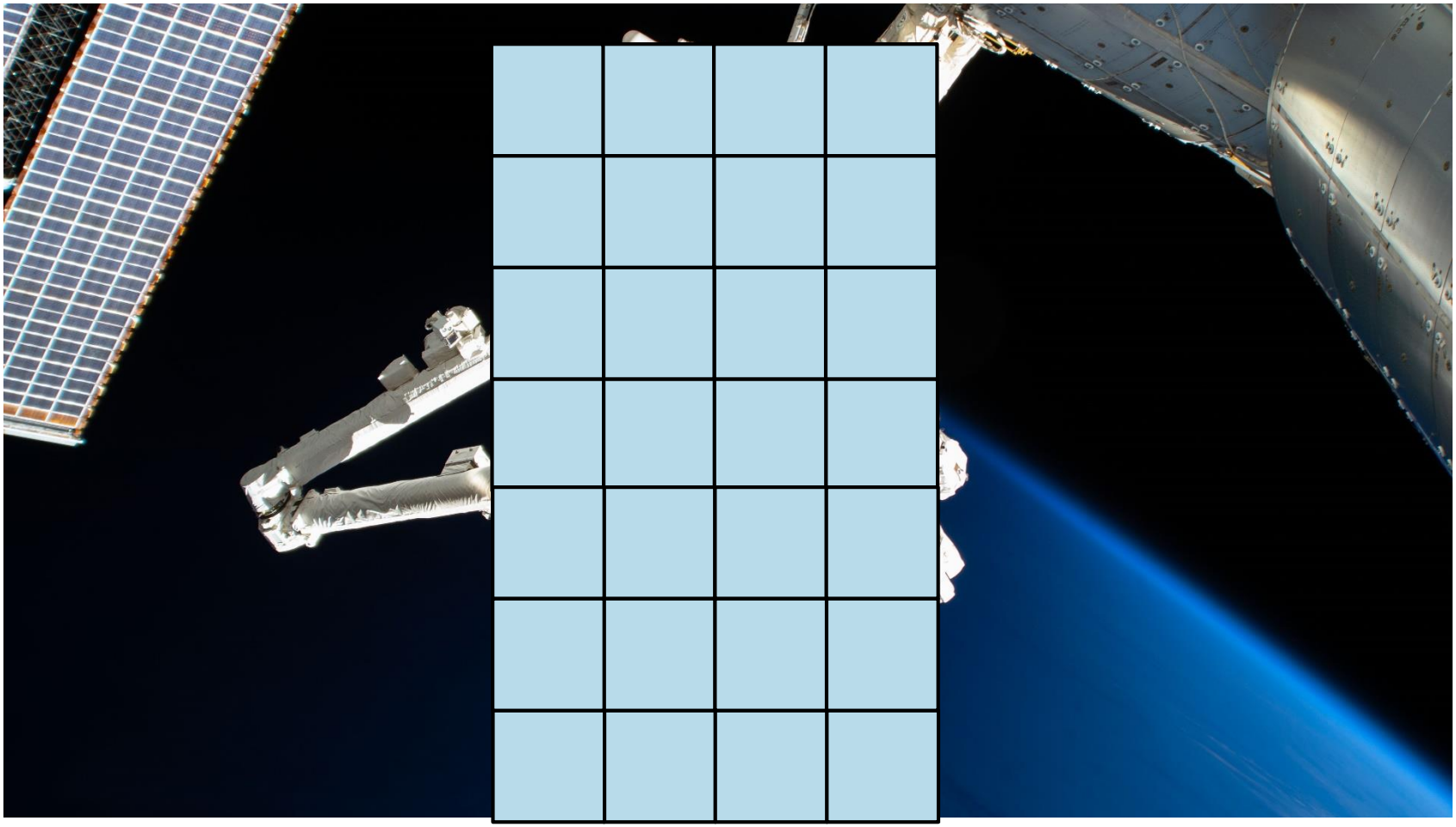
Why should we care about connectivity?



Why should we care about connectivity?



Why should we care about connectivity?

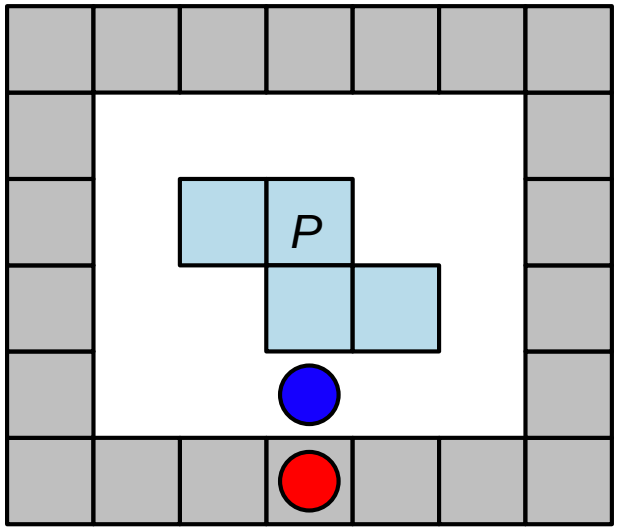


1. Constructing a Bounding Box

Given: A polyomino *P.*

Wanted: A rectangular box that surrounds *P.*

Constraint: The union of all tiles and all robots has to be connected at any time.



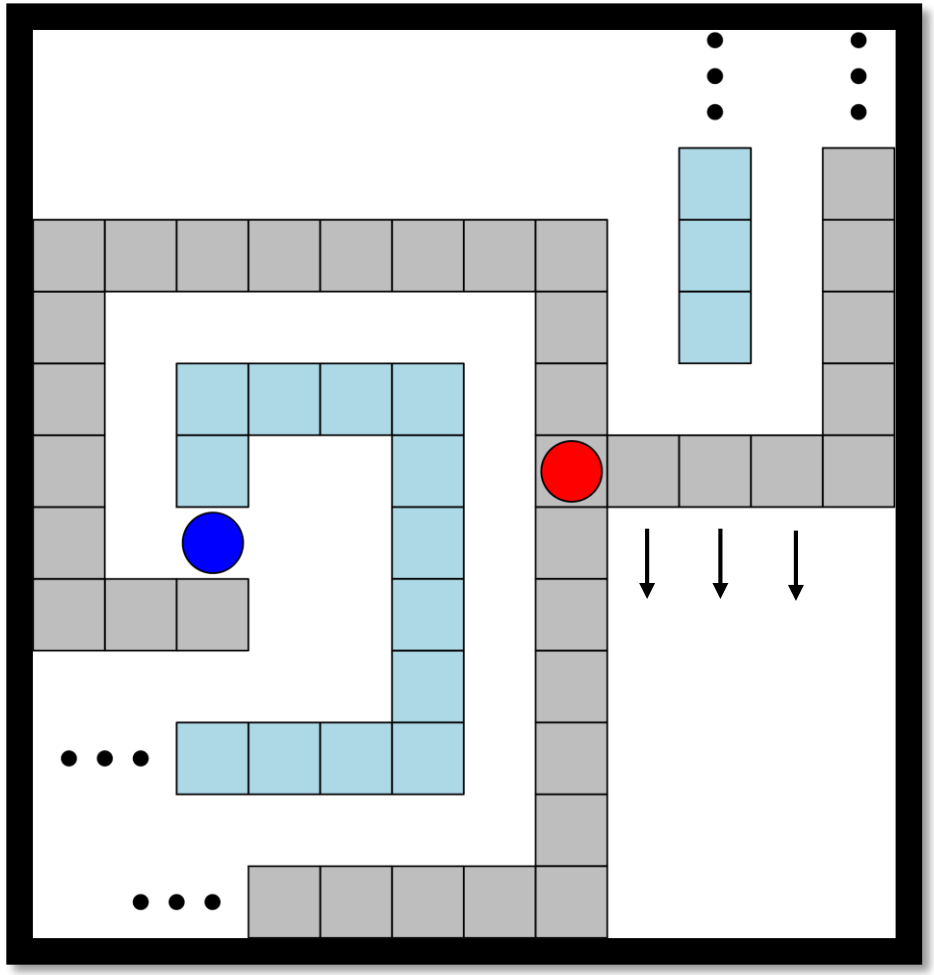
Constructing a Bounding Box

- Two robots (one as a special marker)
- 3 phases:

1. **Finding a start position**

2. **Construction**
   - Clockwise
   - Shifting
     - Conflict on starting edge
   - Hit a bounding box tile

3. **Clean-Up (Finishing)**

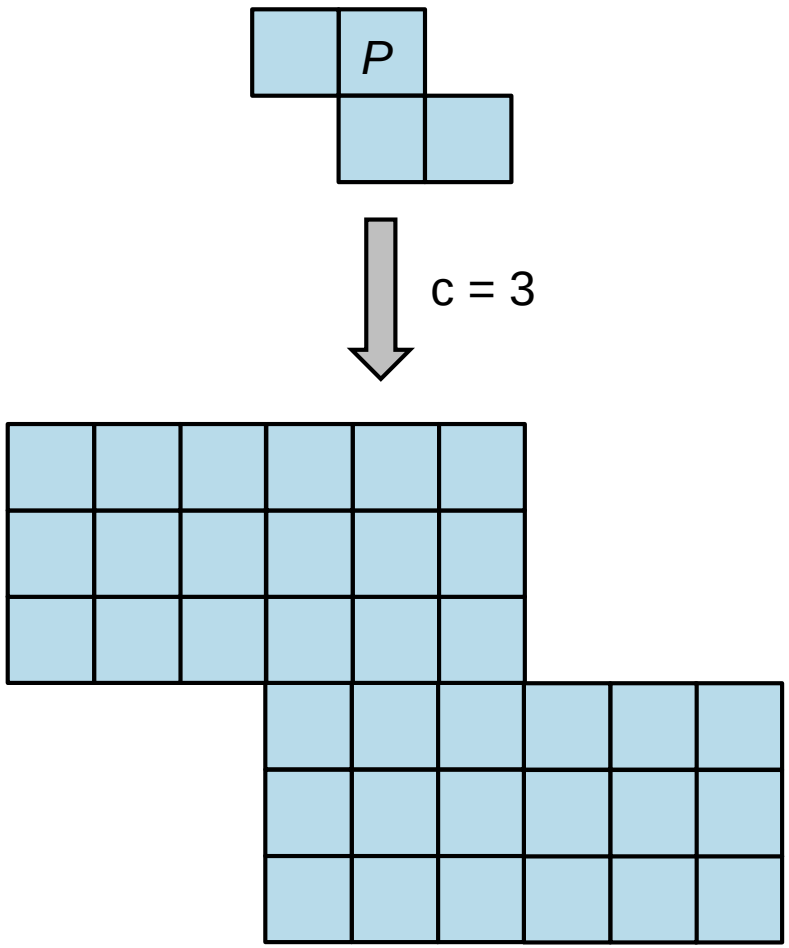


2. Scaling Polyominoes

Given: A polyomino *P,* a constant *c.*

Wanted: Scale every tile of *P* by the constant factor *c.*

Constraint: The union of all tiles and all robots has to be connected at any time.

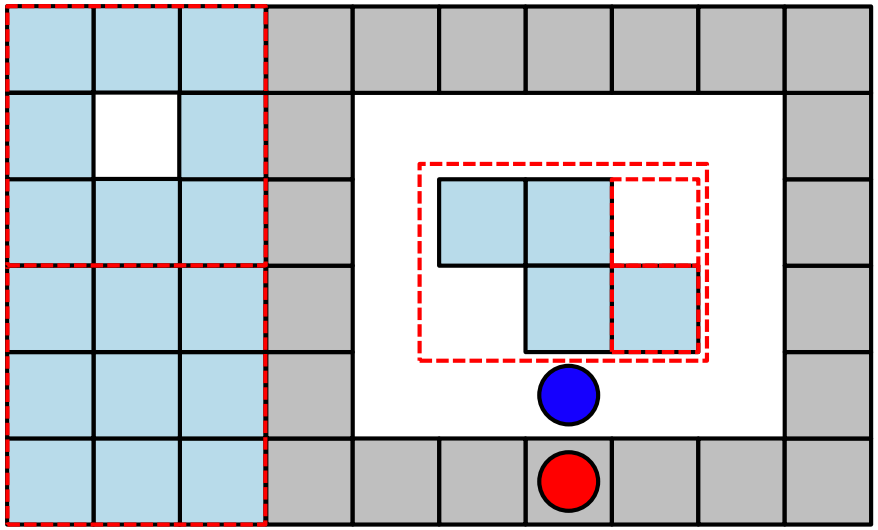


Scaling Polyominoes

- Pre-Step: Bounding box

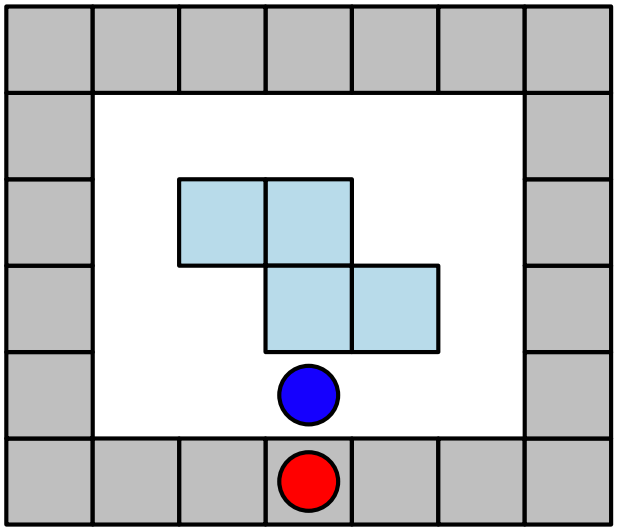
Idea:

- For every node within the bounded area:
  - Place a $c \times c$ square
  - Marker for empty nodes of P



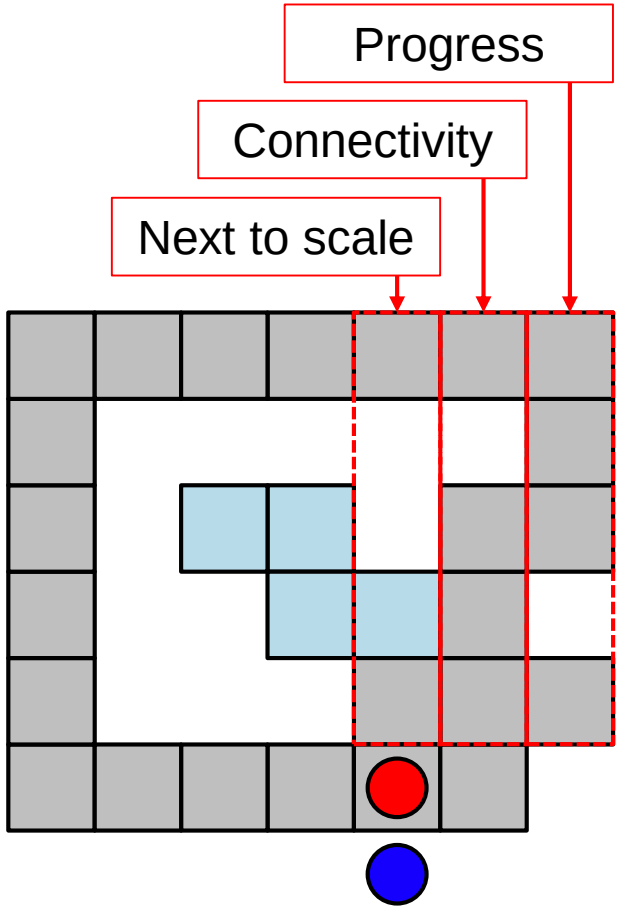
Scaling Polyominoes

1. Preparation



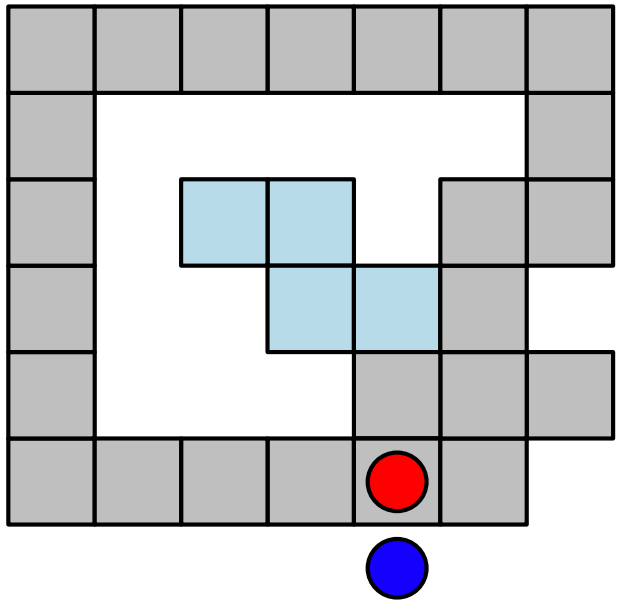
Scaling Polyominoes

1. Preparation



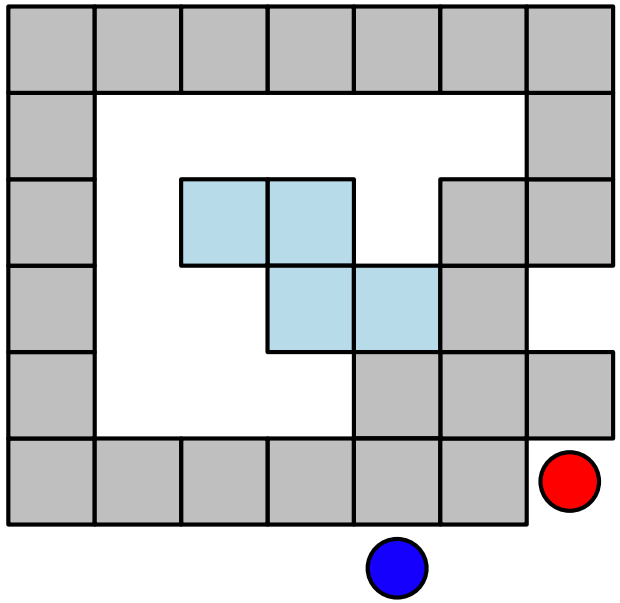
Scaling Polyominoes

1. Preparation → **2. Scaling**



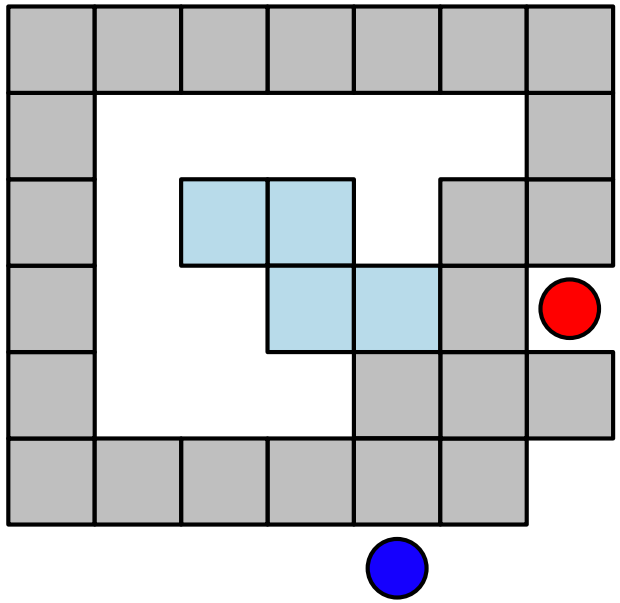
Scaling Polyominoes

1. Preparation → **2. Scaling**



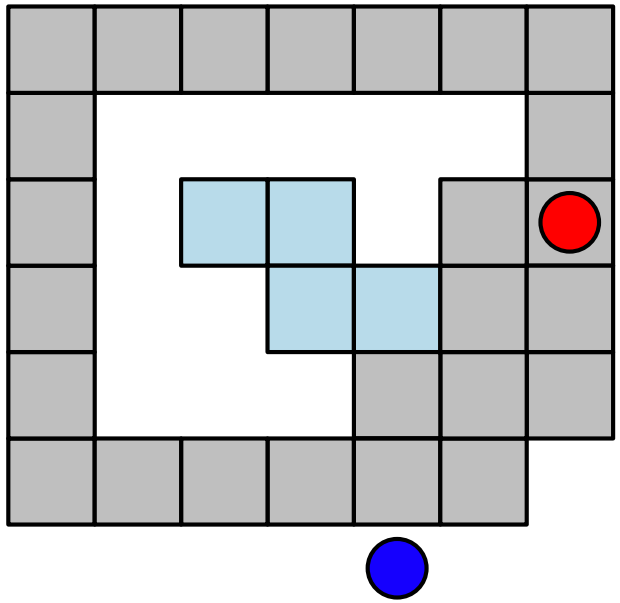
Scaling Polyominoes

1. Preparation → **2. Scaling**



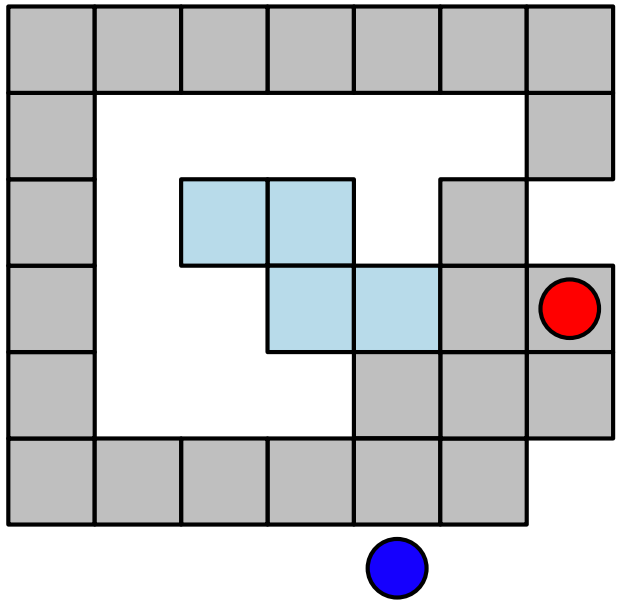
Scaling Polyominoes

1. Preparation → **2. Scaling**



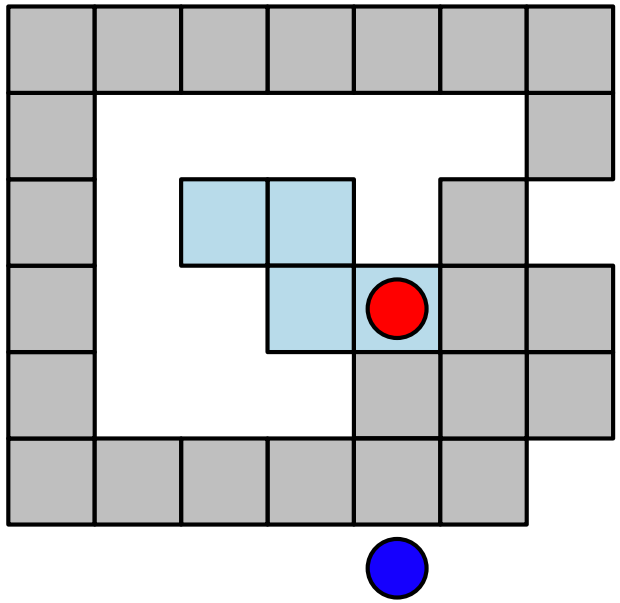
Scaling Polyominoes

1. Preparation → **2. Scaling**



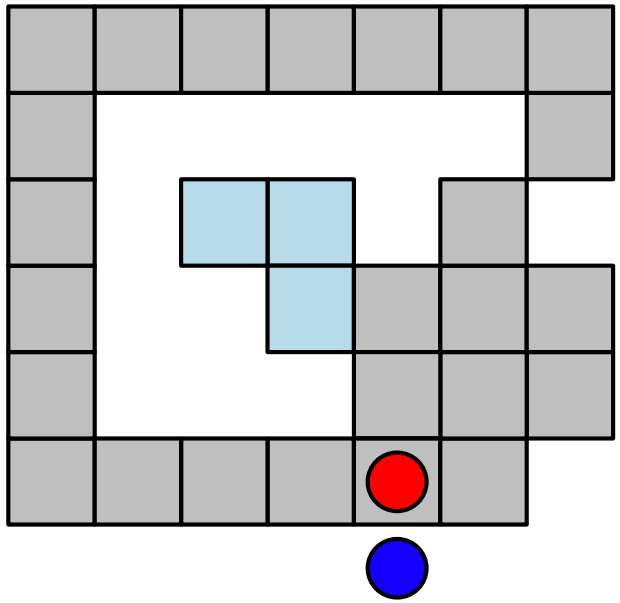
Scaling Polyominoes

1. Preparation → **2. Scaling**



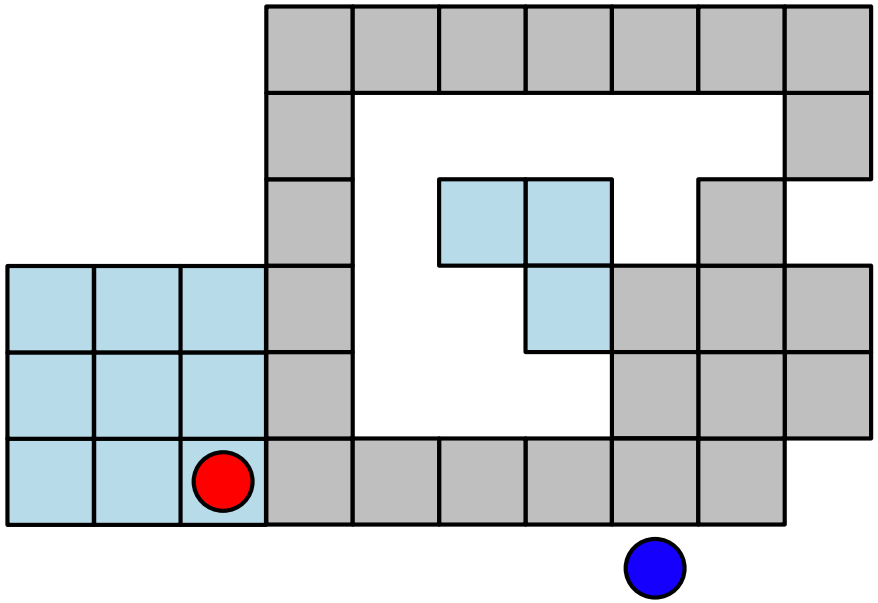
Scaling Polyominoes

1. Preparation → **2. Scaling**



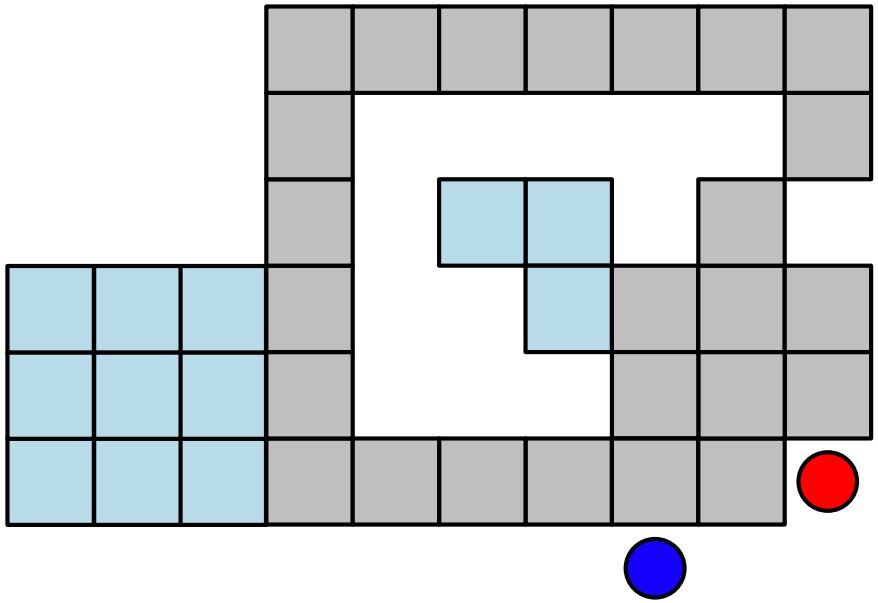
Scaling Polyominoes

1. Preparation → **2. Scaling**



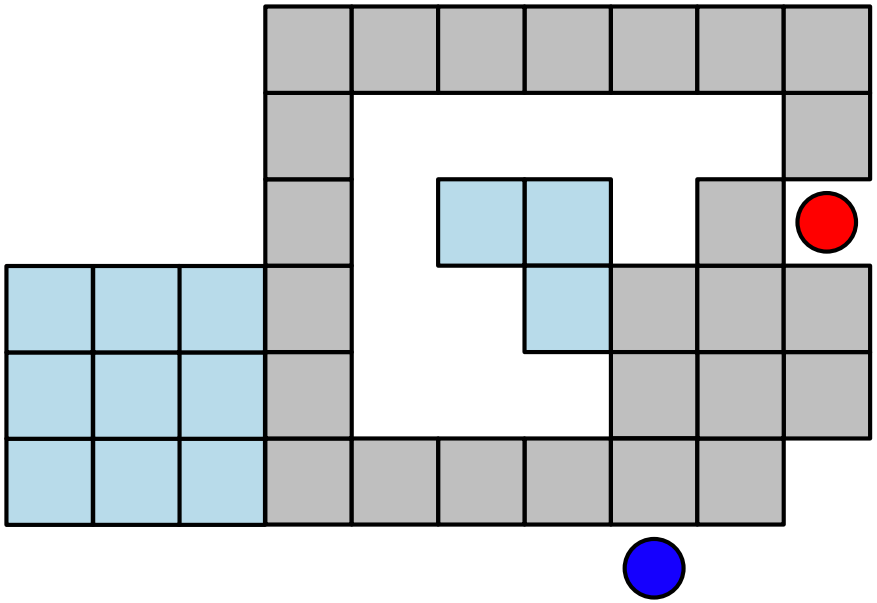
Scaling Polyominoes

1. Preparation → **2. Scaling**



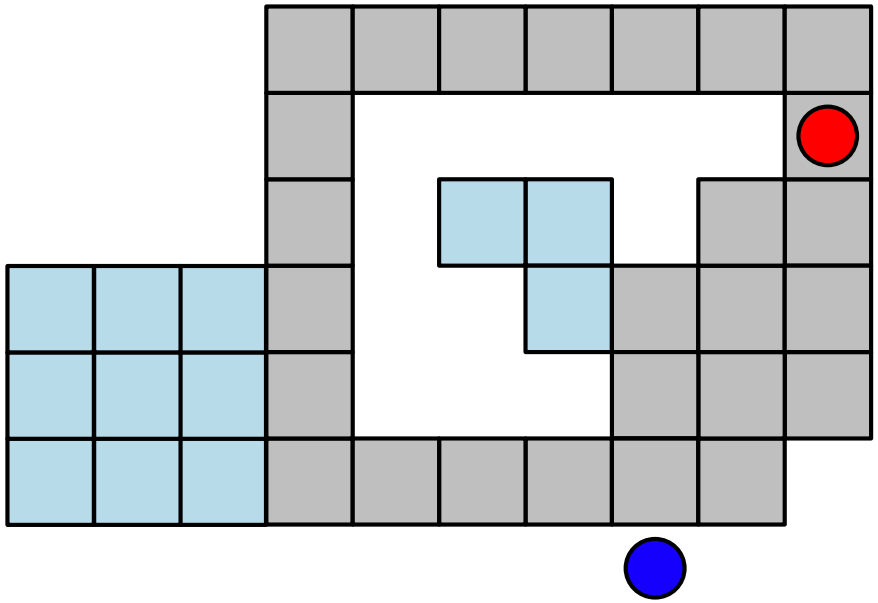
Scaling Polyominoes

1. Preparation → **2. Scaling**



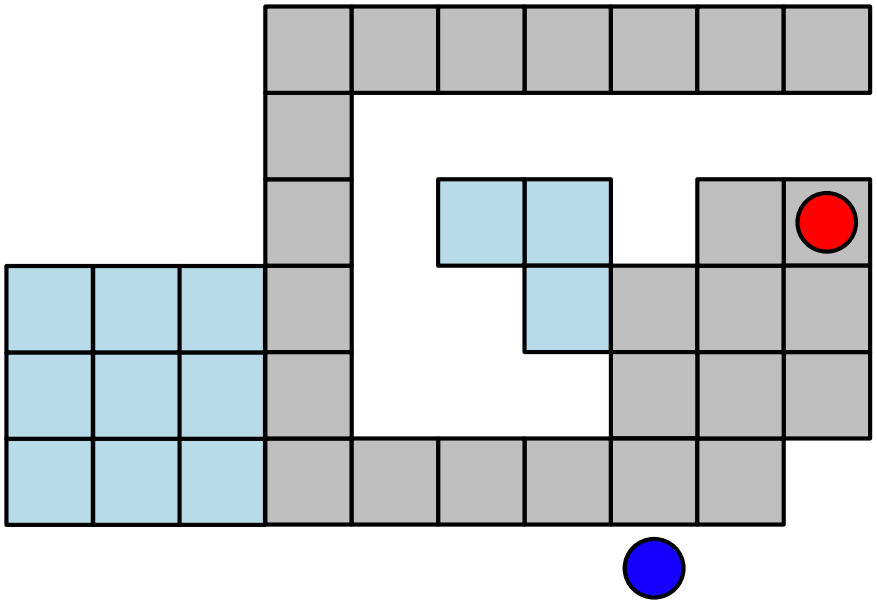
Scaling Polyominoes

1. Preparation → **2. Scaling**



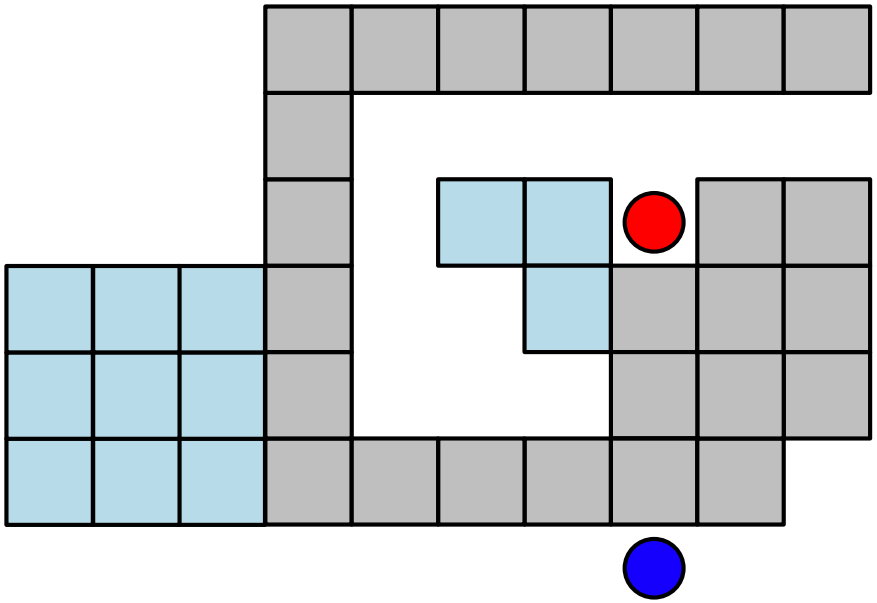
Scaling Polyominoes

1. Preparation → **2. Scaling**



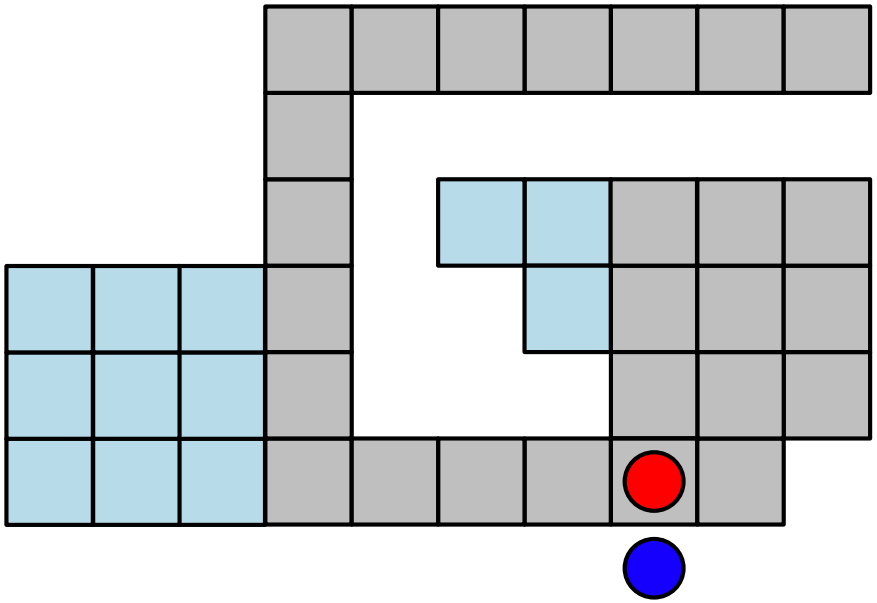
Scaling Polyominoes

1. Preparation → **2. Scaling**



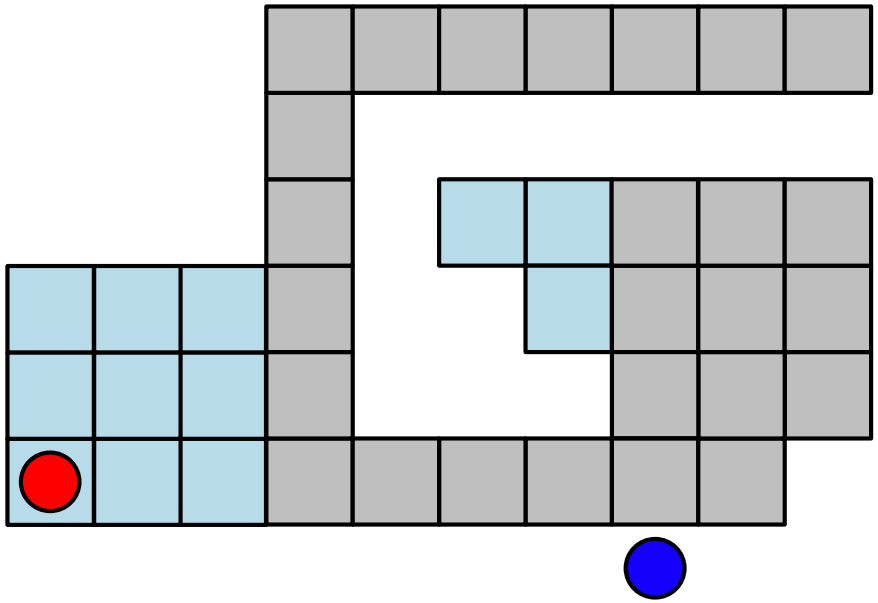
Scaling Polyominoes

1. Preparation → **2. Scaling**



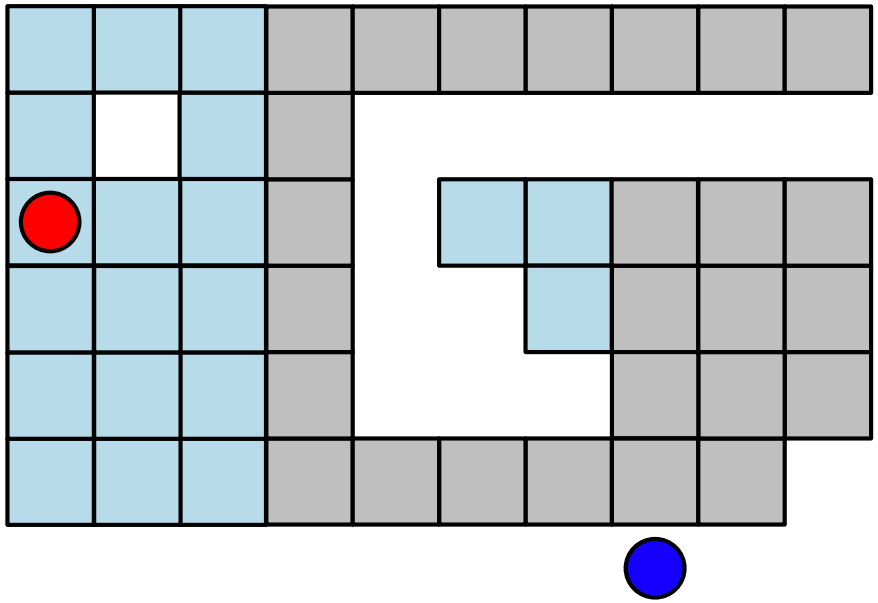
Scaling Polyominoes

1. Preparation → **2. Scaling**



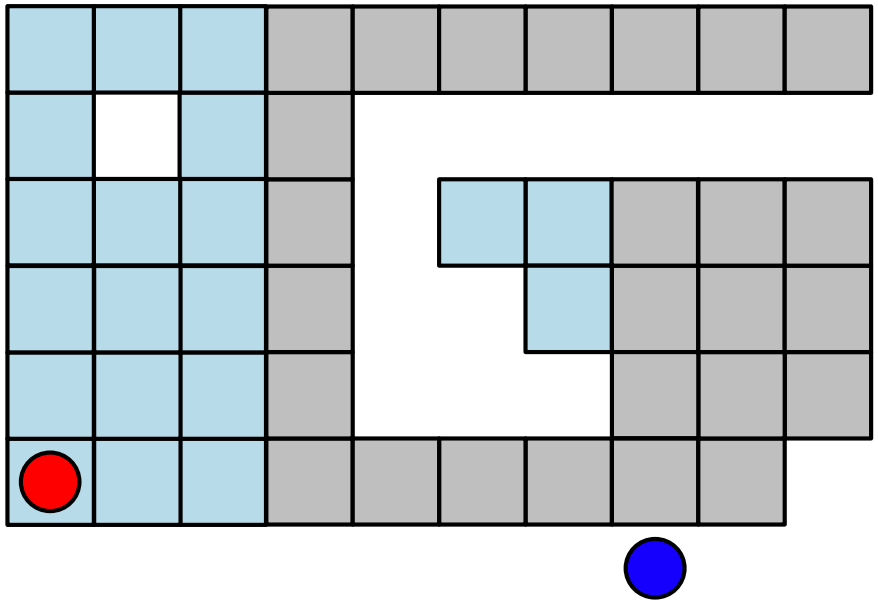
Scaling Polyominoes

1. Preparation → **2. Scaling**



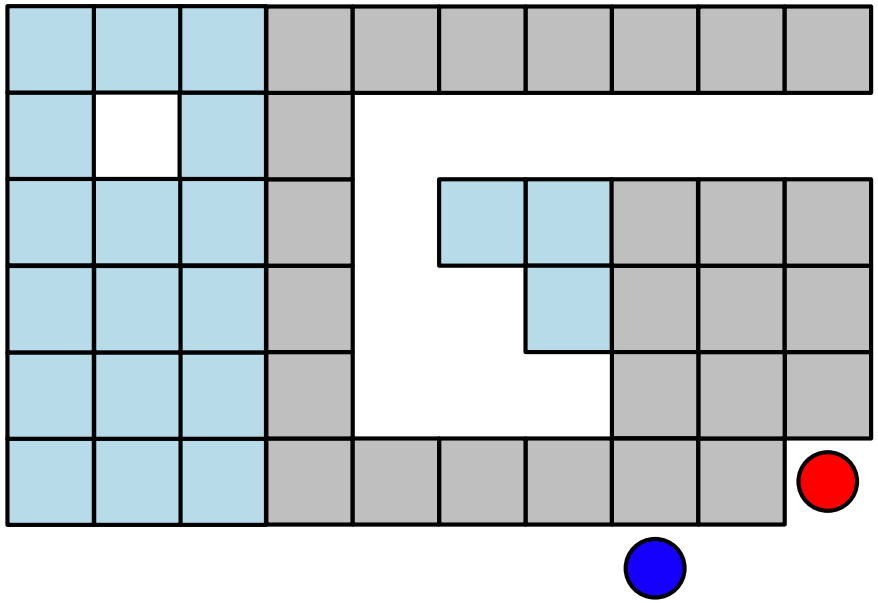
Scaling Polyominoes

1. Preparation → **2. Scaling**



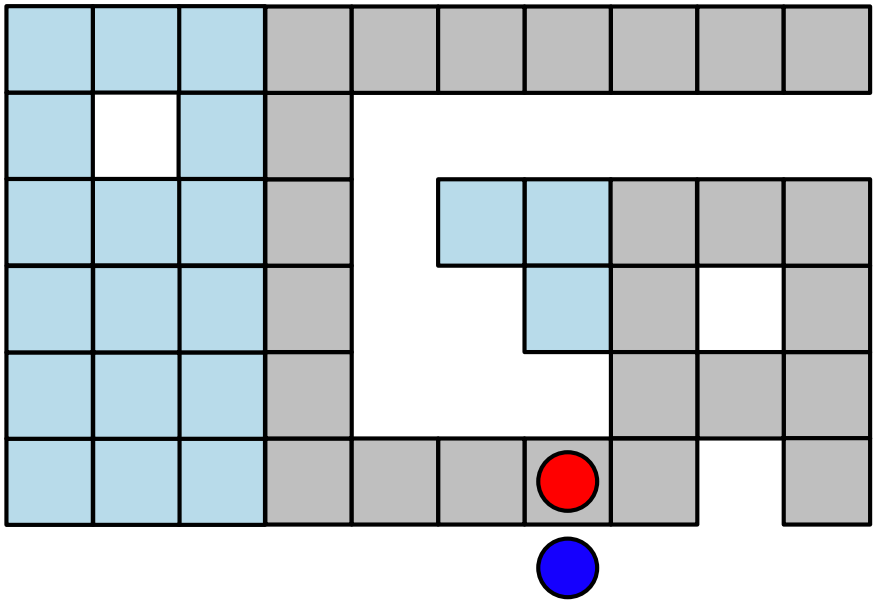
Scaling Polyominoes

1. Preparation → **2. Scaling**



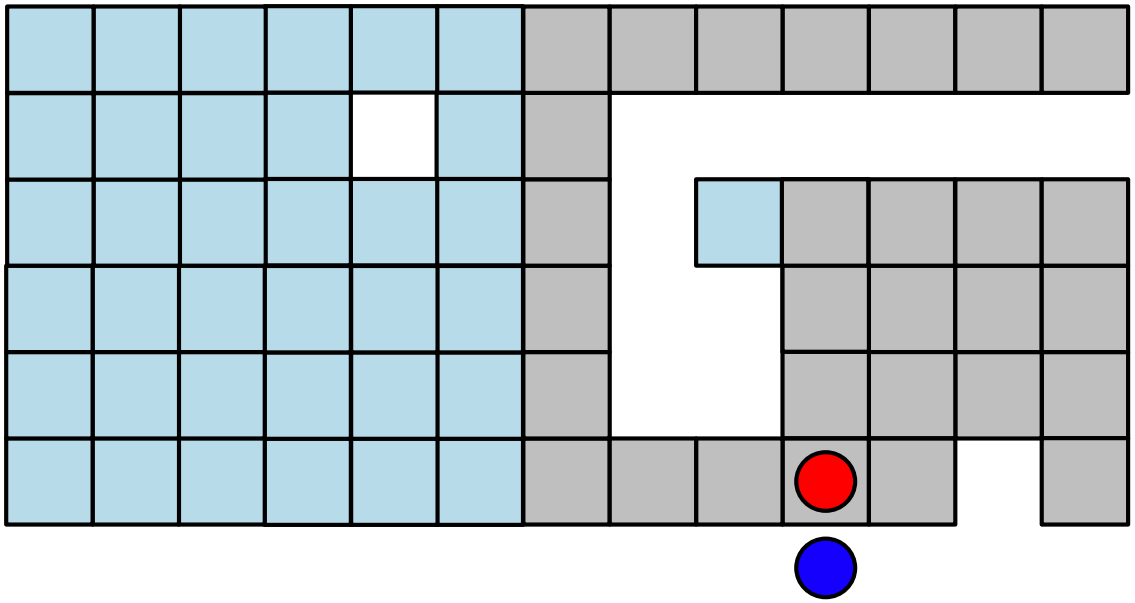
Scaling Polyominoes

1. Preparation → **2. Scaling**



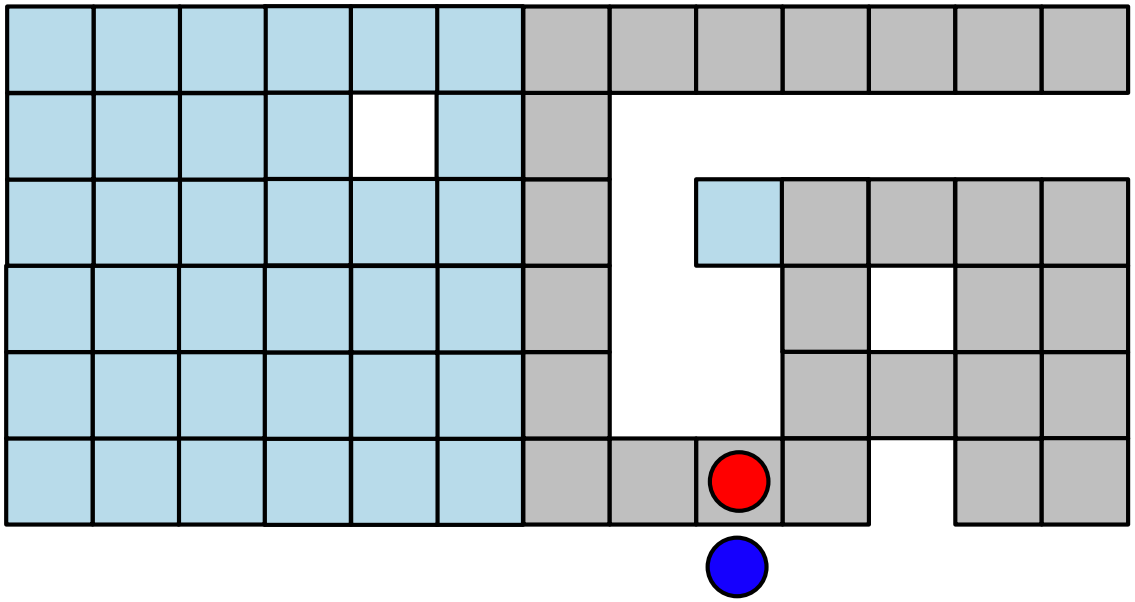
Scaling Polyominoes

1. Preparation → **2. Scaling**



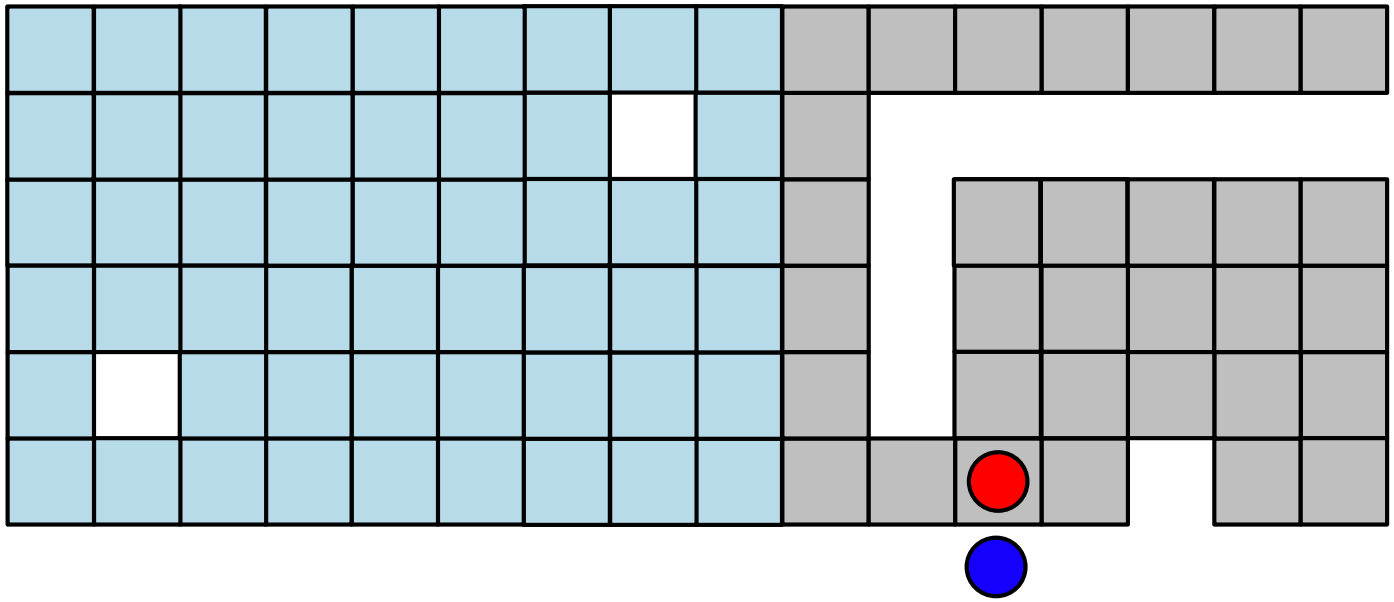
Scaling Polyominoes

1. Preparation → **2. Scaling**



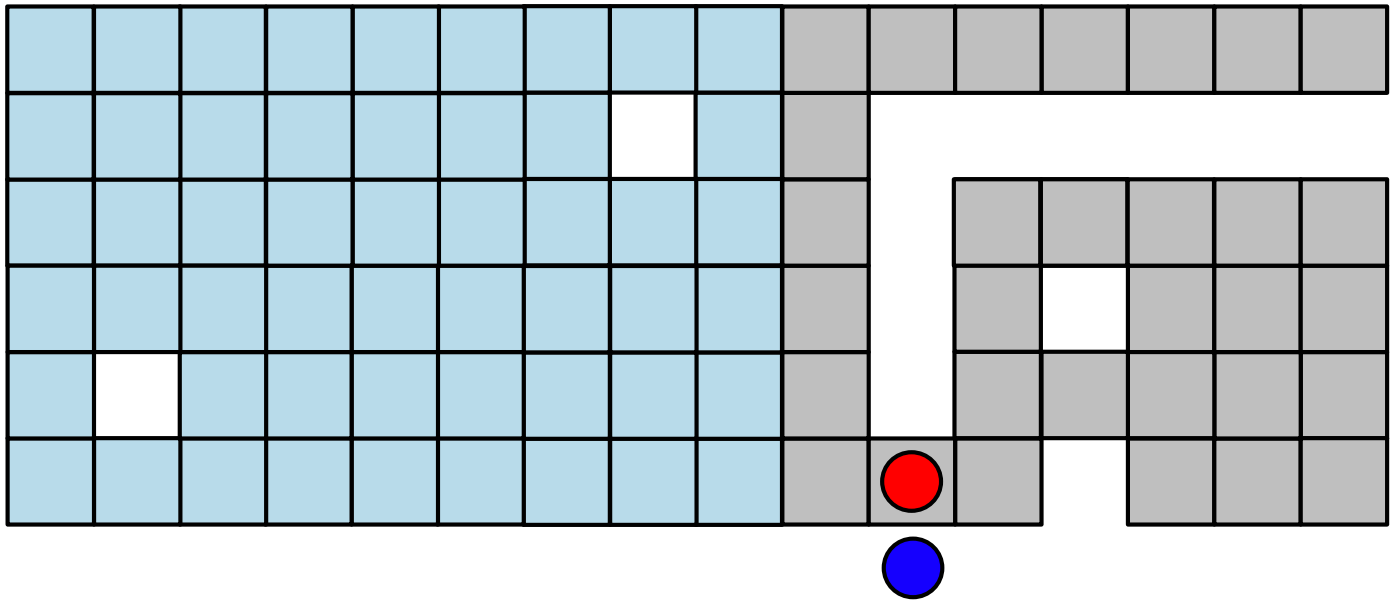
Scaling Polyominoes

1. Preparation → **2. Scaling**



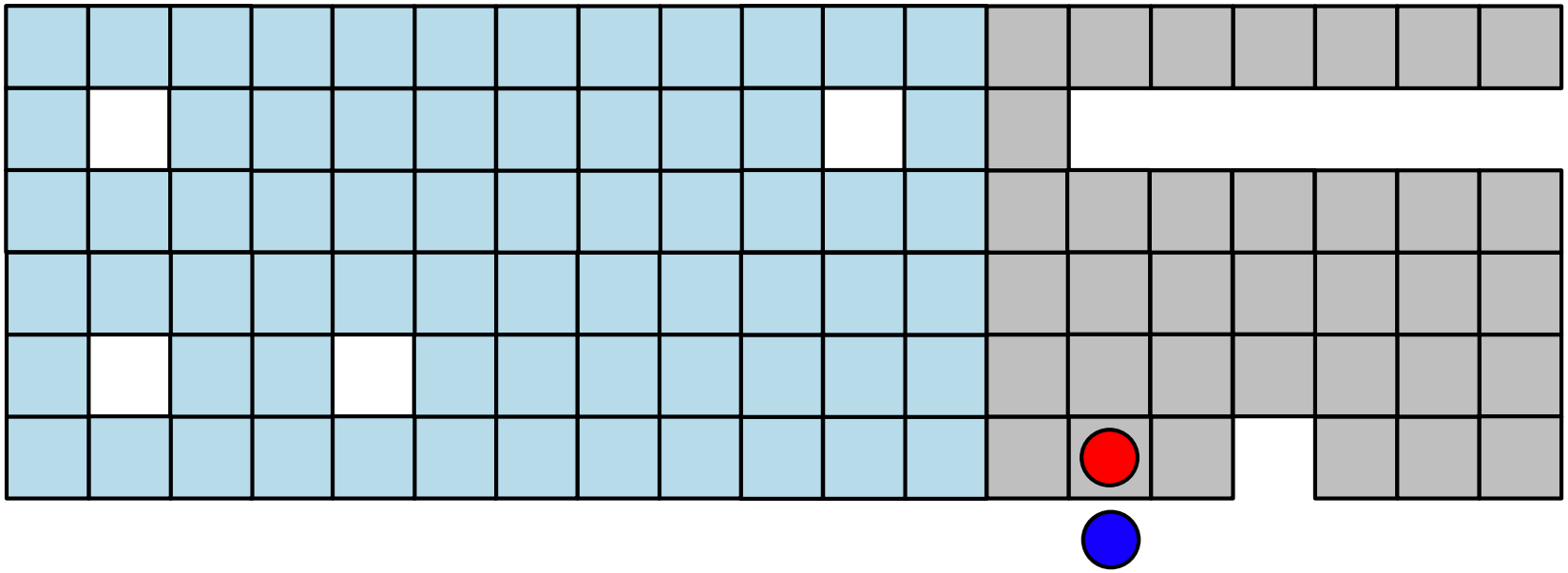
Scaling Polyominoes

1. Preparation → **2. Scaling**



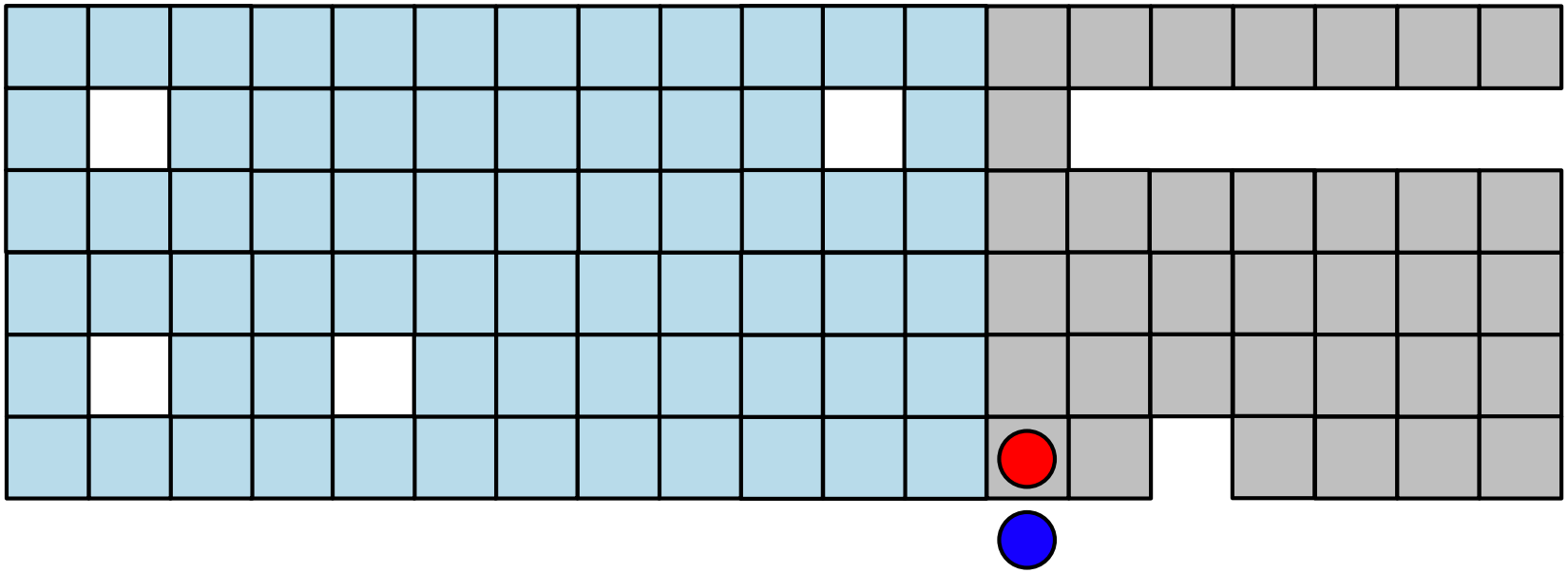
Scaling Polyominoes

1. Preparation → **2. Scaling**



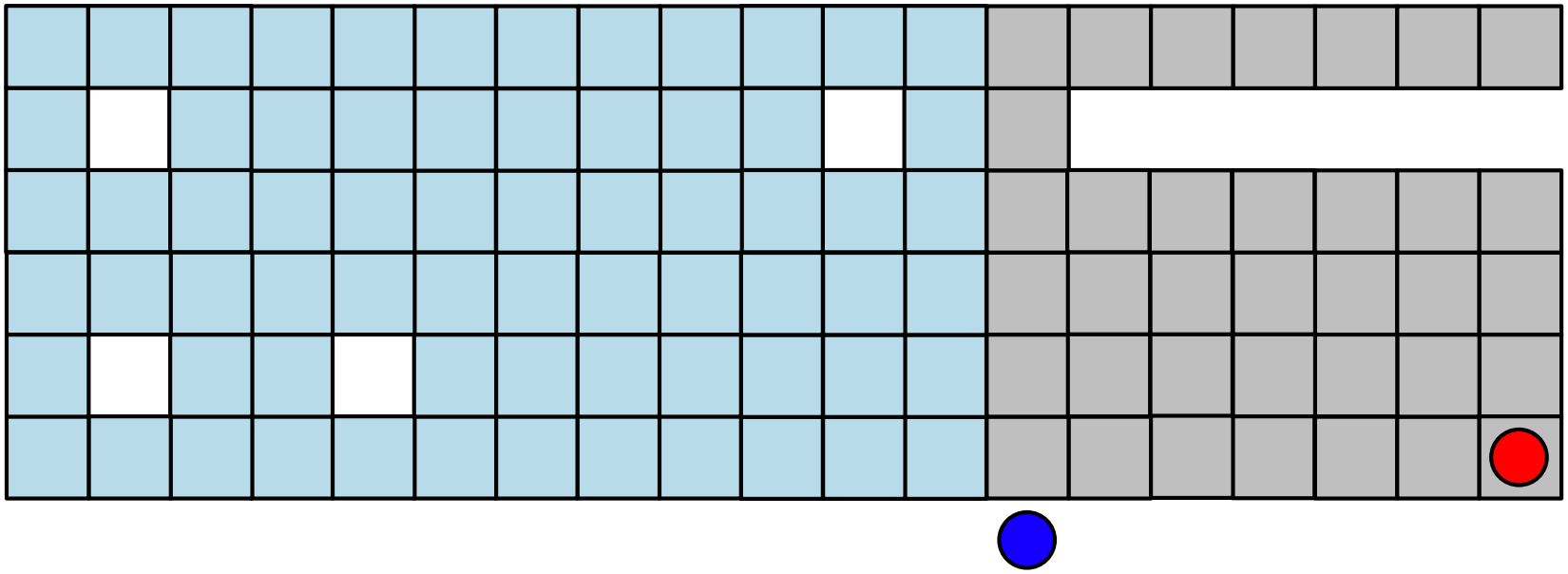
Scaling Polyominoes

1. Preparation → **2. Scaling**



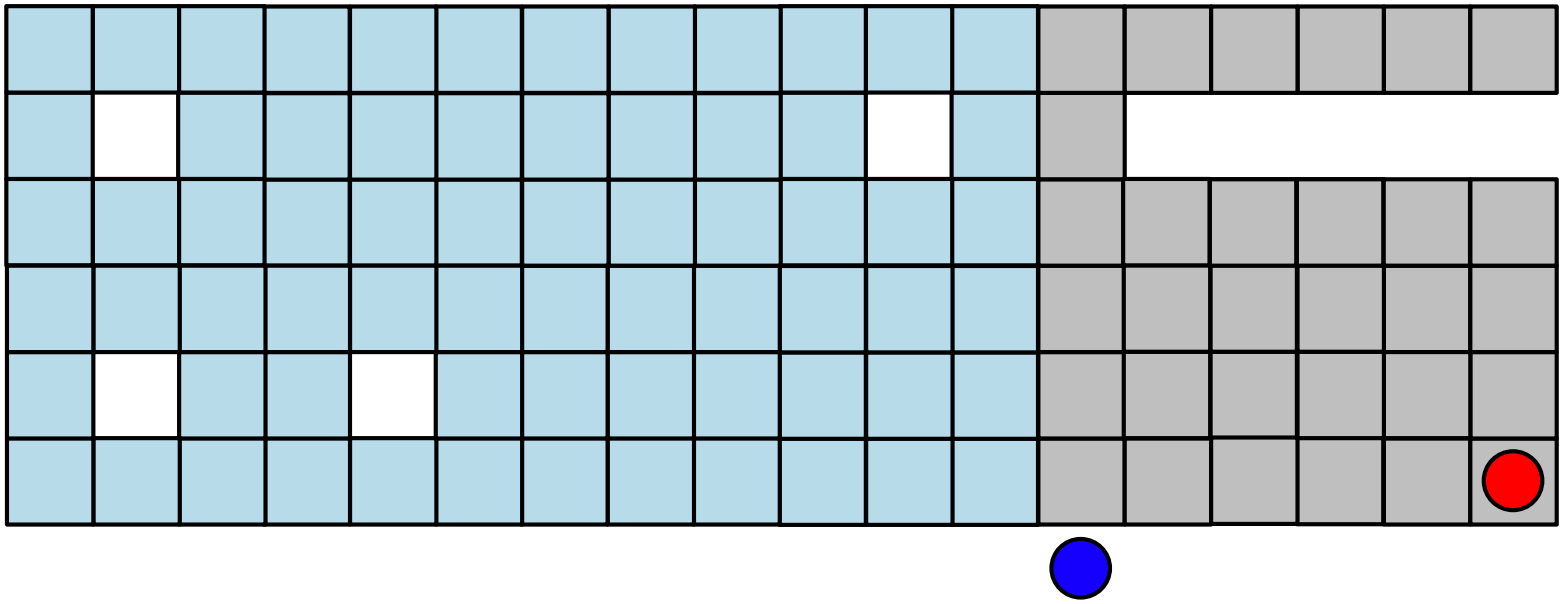
Scaling Polyominoes

1. Preparation → 2. Scaling → **3. Clean-Up**



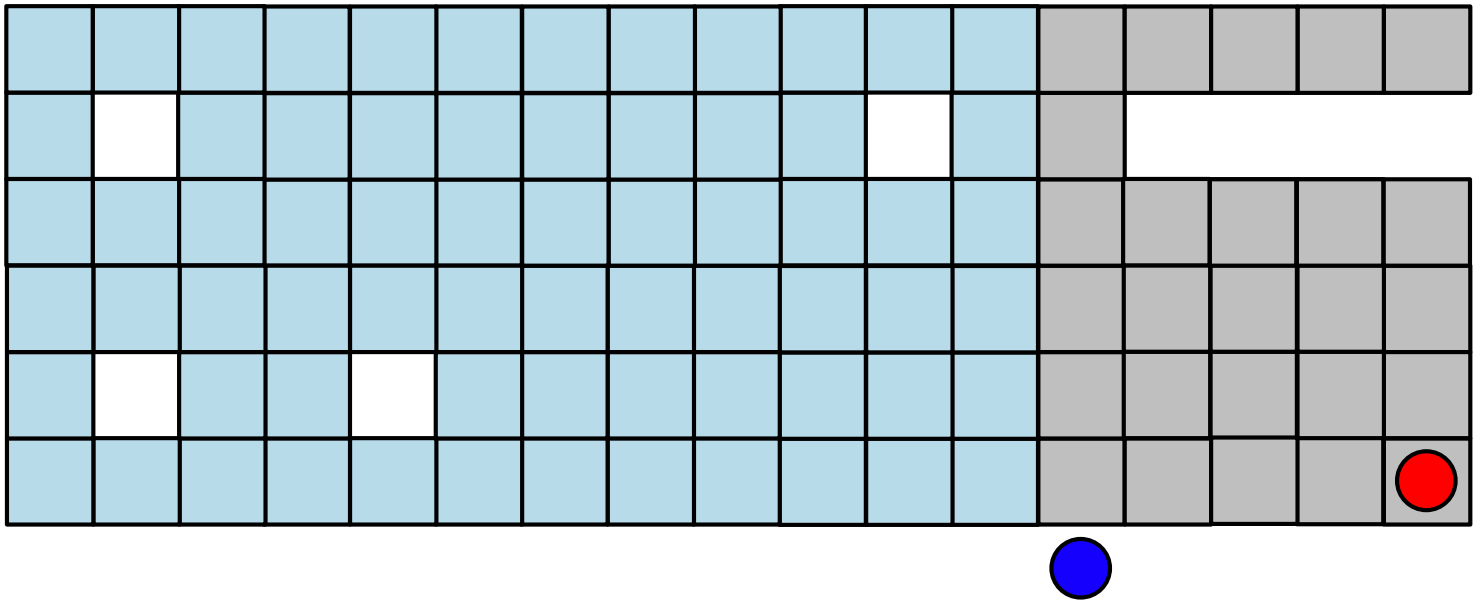
Scaling Polyominoes

1. Preparation → 2. Scaling → **3. Clean-Up**



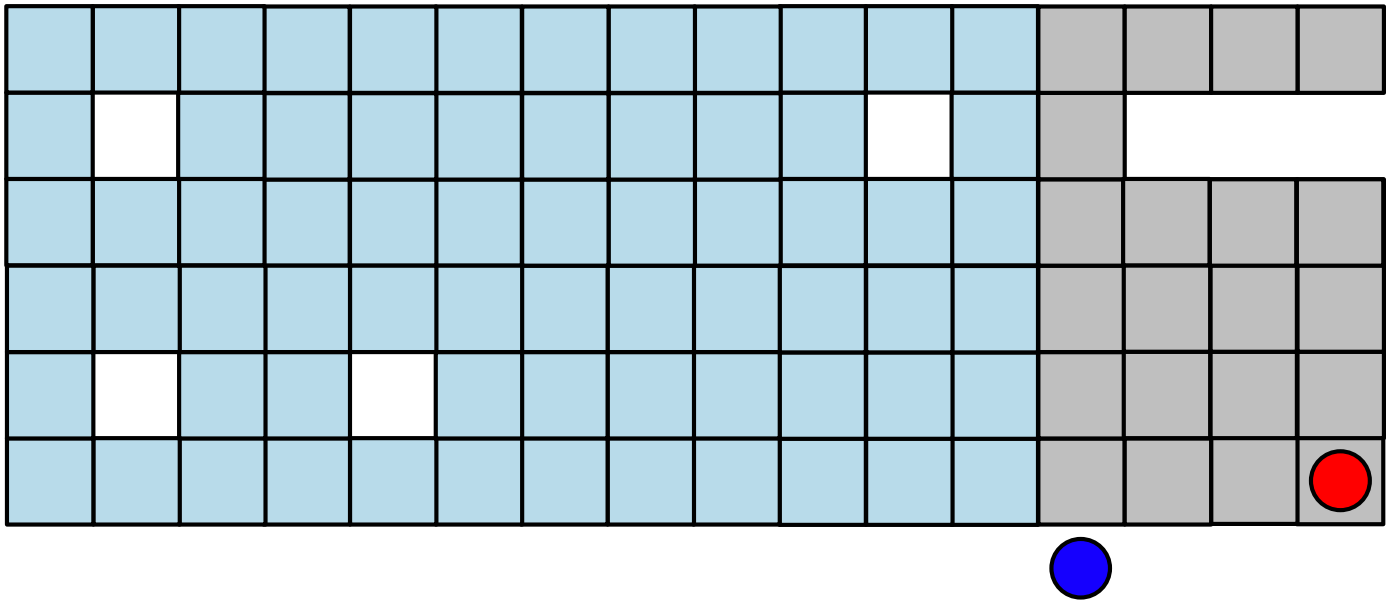
Scaling Polyominoes

1. Preparation → 2. Scaling → **3. Clean-Up**



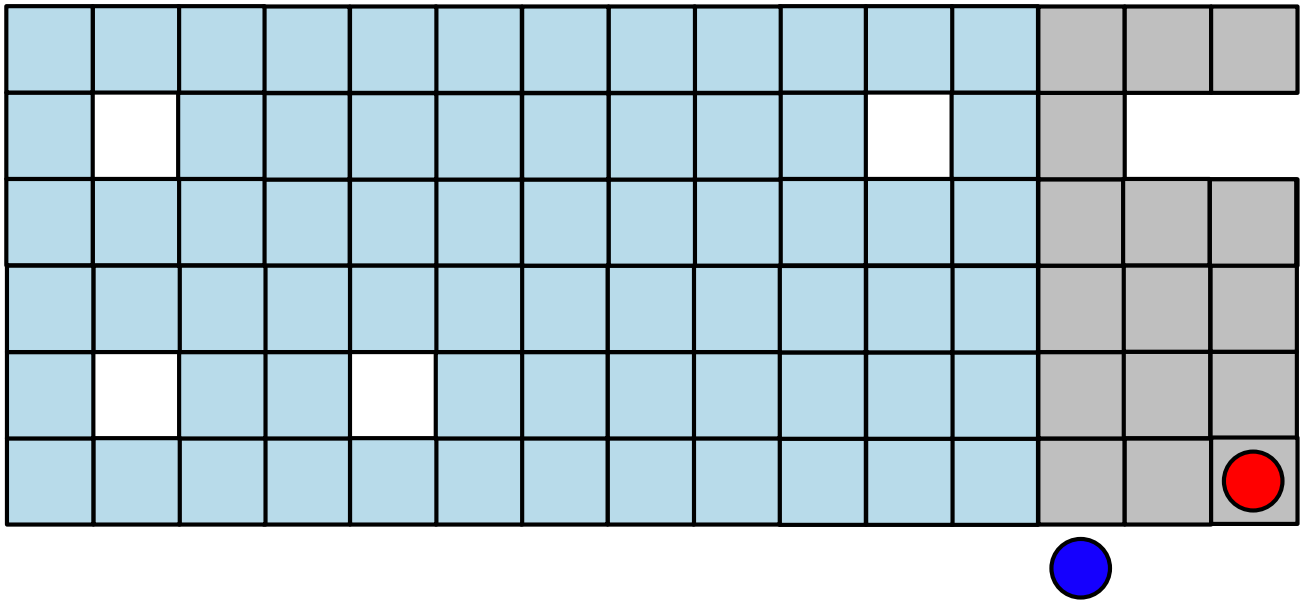
Scaling Polyominoes

1. Preparation → 2. Scaling → **3. Clean-Up**



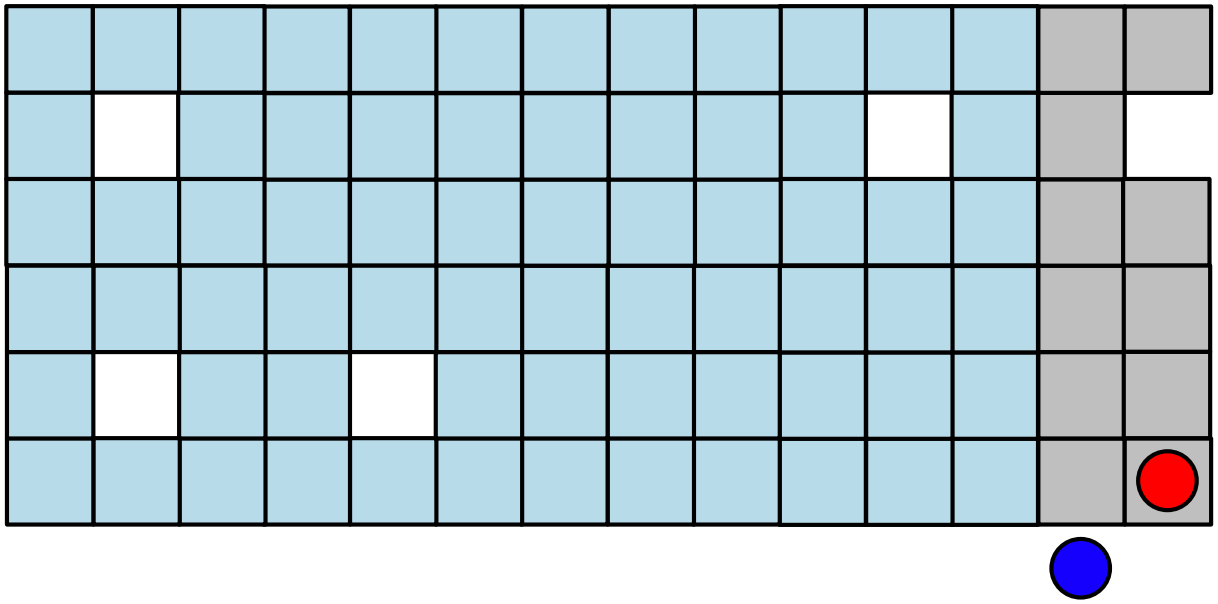
Scaling Polyominoes

1. Preparation → 2. Scaling → **3. Clean-Up**



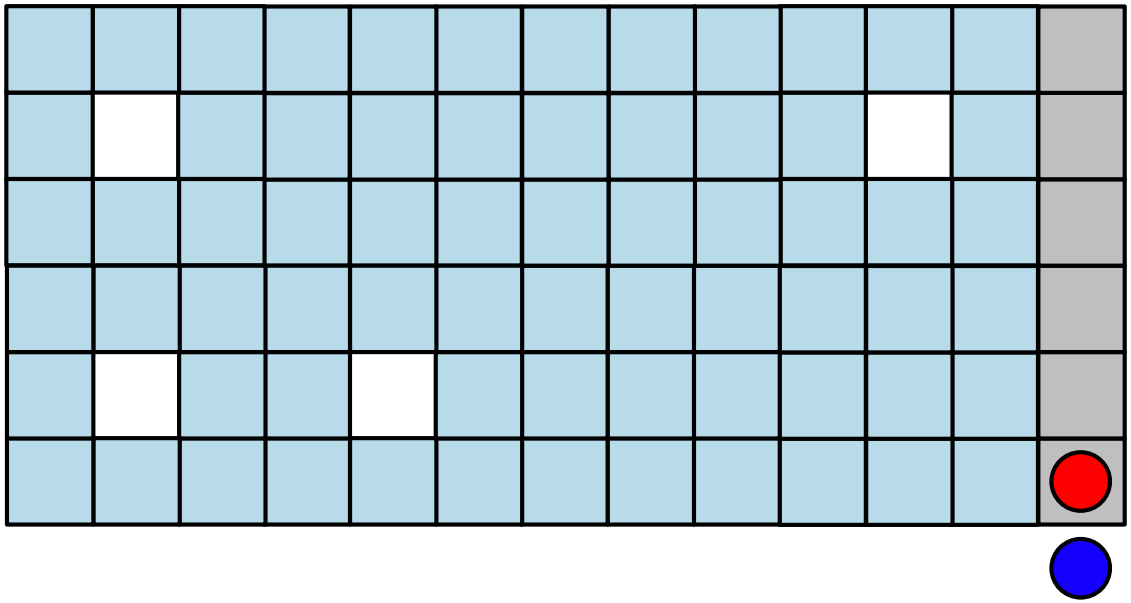
Scaling Polyominoes

1. Preparation → 2. Scaling → **3. Clean-Up**



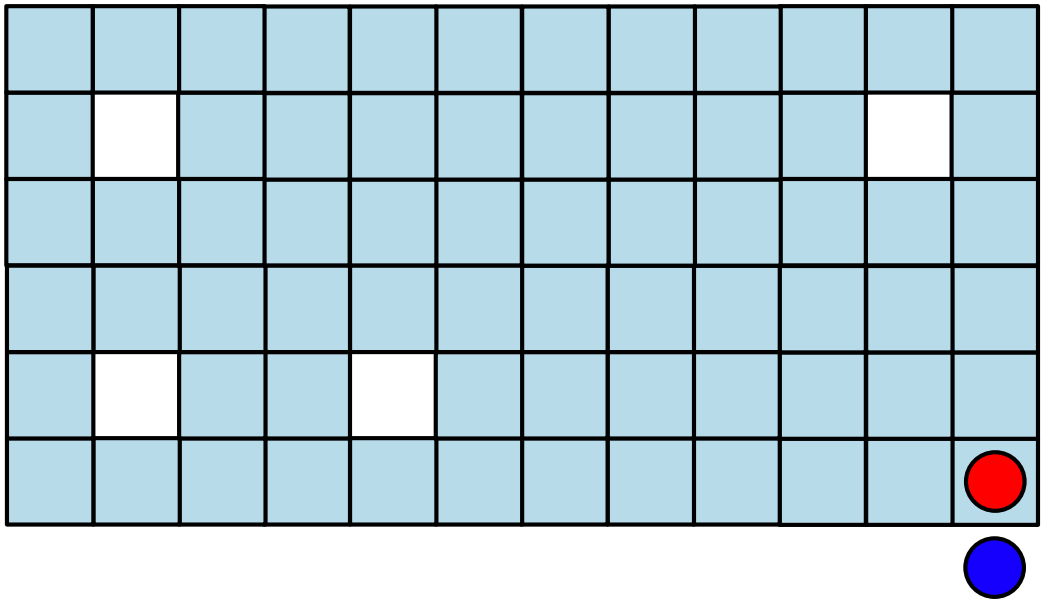
Scaling Polyominoes

1. Preparation → 2. Scaling → **3. Clean-Up**



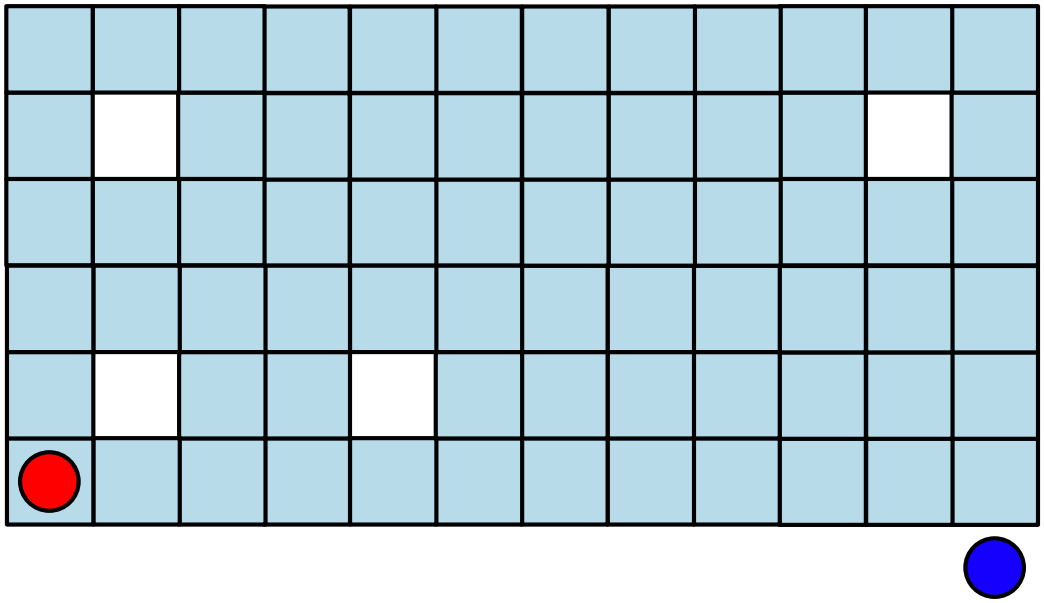
Scaling Polyominoes

1. Preparation → 2. Scaling → **3. Clean-Up**



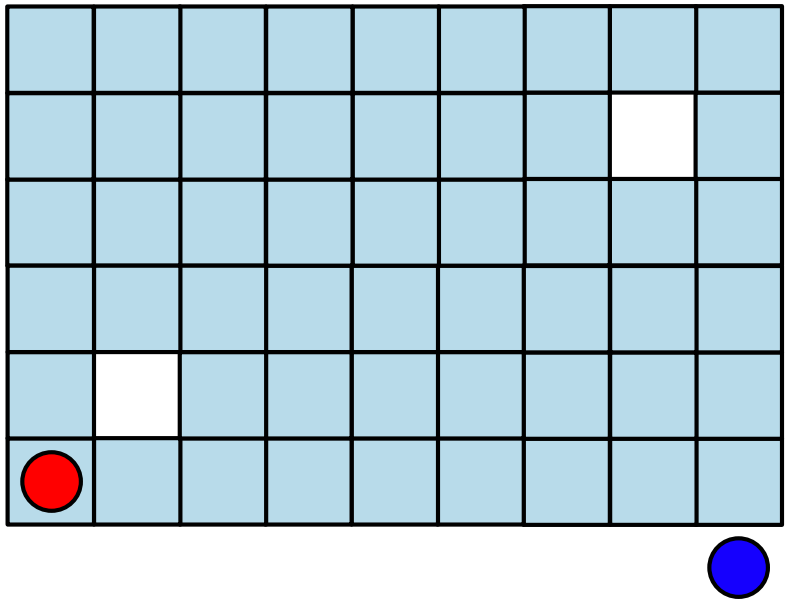
Scaling Polyominoes

1. Preparation → 2. Scaling → **3. Clean-Up**



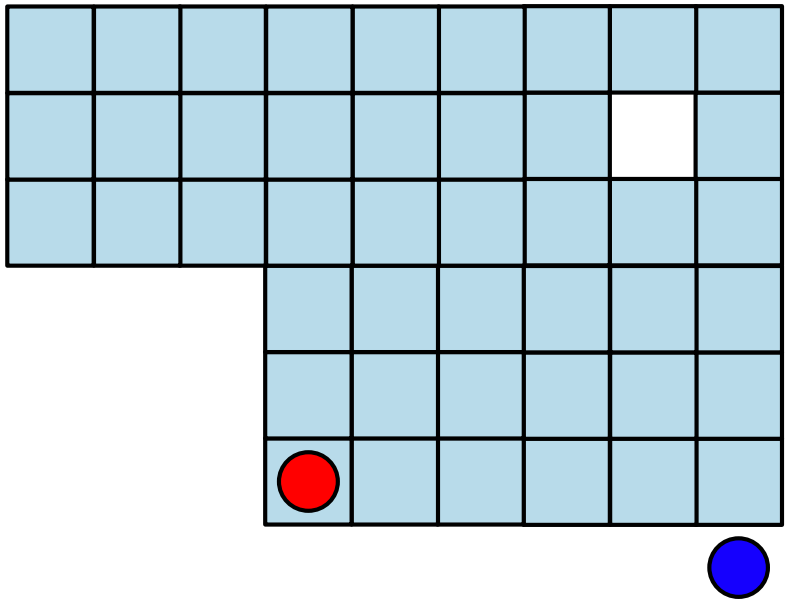
Scaling Polyominoes

1. Preparation → 2. Scaling → **3. Clean-Up**



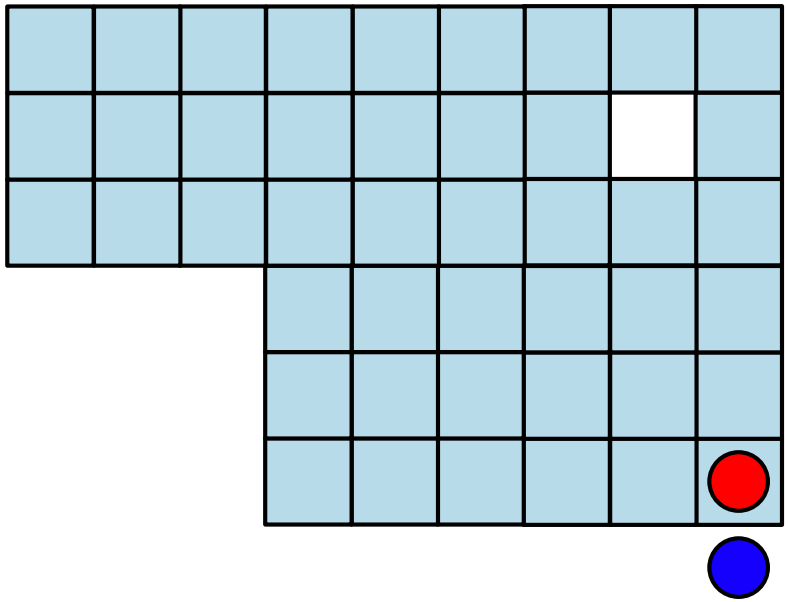
Scaling Polyominoes

1. Preparation → 2. Scaling → **3. Clean-Up**



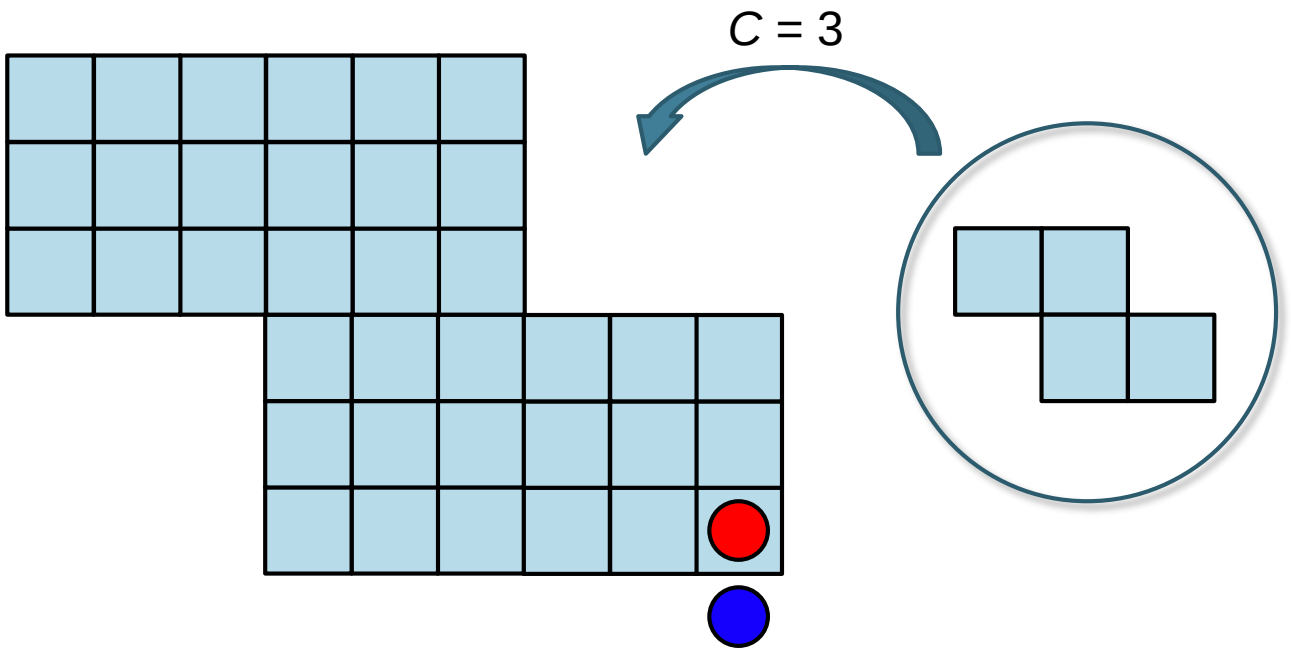
Scaling Polyominoes

1. Preparation → 2. Scaling → **3. Clean-Up**



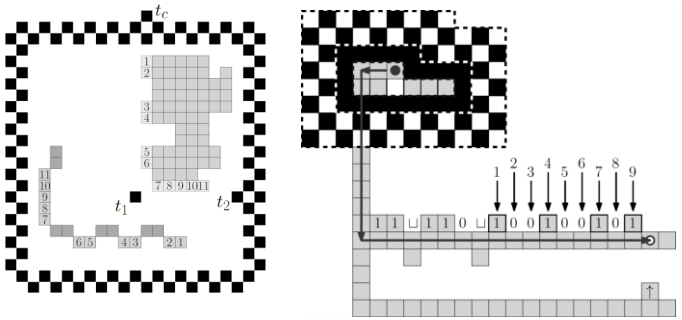
Scaling Polyominoes

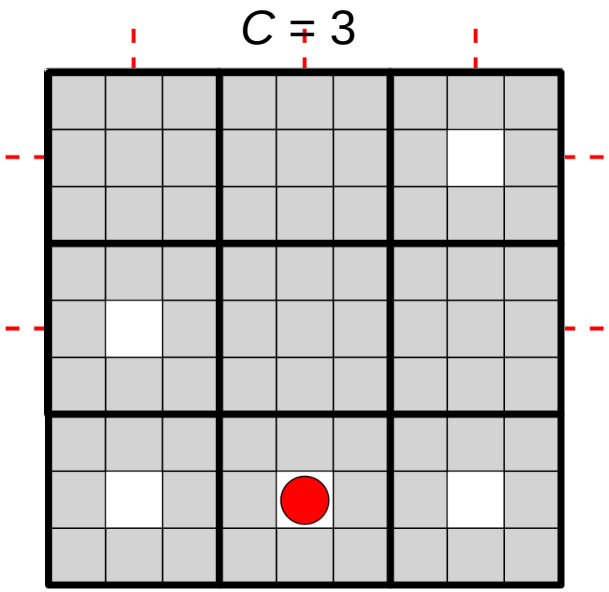
1. Preparation → 2. Scaling → **3. Clean-Up**



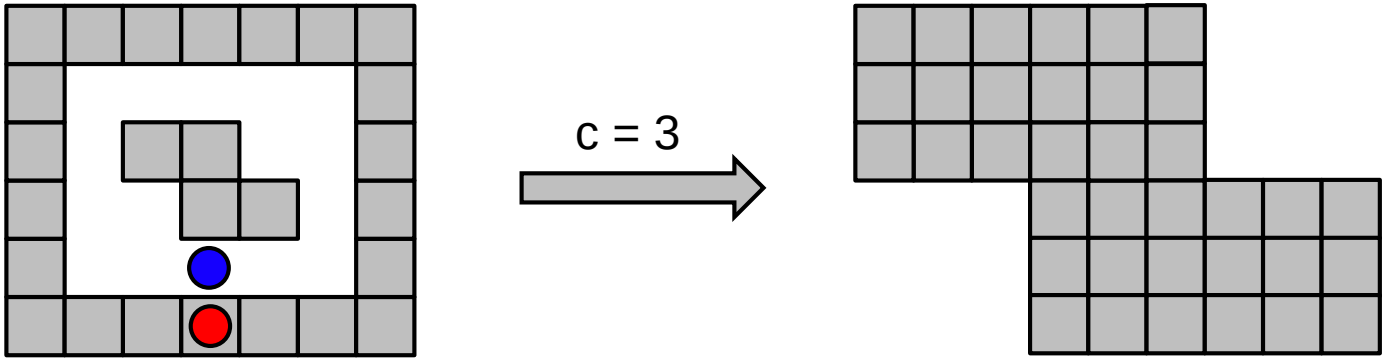
Adapting Algorithms

- Given any Robots-on-Tiles algorithm – A
- We construct an algorithm A’ that ensures connectivity:
  1. Scale P by a factor c
  2. c steps in A’ for every step in A
  3. Fill rows/columns with $c \times c$ segments, when it is first entered.



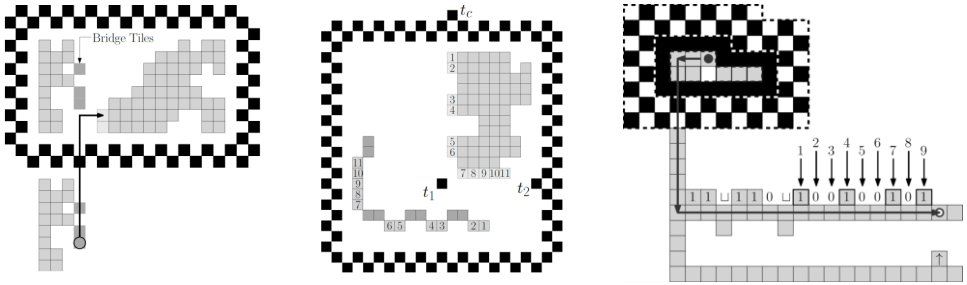


Summary



Bounding Box

Scaling



Adapting algorithms to fulfill connectivity constraints

Proof of Concept