On the Transformation Capability of Feasible Mechanisms for Programmable Matter

Othon Michail George Skretas Paul G. Spirakis

Department of Computer Science, University of Liverpool, UK
Computer Technology Institute and Press "Diophantus" (CTI), Patras, Greece
Computer Engineering and Informatics Department, University of Patras, Greece

44th International Colloquium on Automata, Languages, and Programming (ICALP)
July 10-14, 2017
Warsaw, Poland

Programmable Matter

- Any type of matter that can algorithmically change its physical properties
  - shape, color, strength, connectivity, stiffness, conductivity, ...
- Change/Transformation is the result of executing an underlying program
  - centralized algorithm, or
  - distributed protocol

Vision:

- Materials that can be programmed and controlled, integrating actuation and sensing capabilities
- Thus, materials that are "smart" and able to adapt to changing conditions

Programmable Matter

- Any type of matter that can algorithmically change its physical properties
  - shape, color, strength, connectivity, stiffness, conductivity, ...
- Change/Transformation is the result of executing an underlying program
  - centralized algorithm, or
  - distributed protocol

Vision:

- Materials that can be programmed and controlled, integrating actuation and sensing capabilities
- Thus, materials that are "smart" and able to adapt to changing conditions

Programmable Matter

- Any type of matter that can algorithmically change its physical properties
  - shape, color, strength, connectivity, stiffness, conductivity, ...
- Change/Transformation is the result of executing an underlying program
  - centralized algorithm, or
  - distributed protocol

Vision:

- Materials that can be programmed and controlled, integrating actuation and sensing capabilities
- Thus, materials that are "smart" and able to adapt to changing conditions

Programmable Matter

- Any type of matter that can algorithmically change its physical properties
  - shape, color, strength, connectivity, stiffness, conductivity, ...
- Change/Transformation is the result of executing an underlying program
  - centralized algorithm, or
  - distributed protocol

Vision:

- Materials that can be programmed and controlled, integrating actuation and sensing capabilities
- Thus, materials that are "smart" and able to adapt to changing conditions

Programmable Matter

- Any type of matter that can algorithmically change its physical properties
  - shape, color, strength, connectivity, stiffness, conductivity, ...
- Change/Transformation is the result of executing an underlying program
  - centralized algorithm, or
  - distributed protocol

Vision:

- Materials that can be programmed and controlled, integrating actuation and sensing capabilities
- Thus, materials that are "smart" and able to adapt to changing conditions

Programmable Matter

- Any type of matter that can algorithmically change its physical properties
  - shape, color, strength, connectivity, stiffness, conductivity, ...
- Change/Transformation is the result of executing an underlying program
  - centralized algorithm, or
  - distributed protocol

Vision:

- Materials that can be programmed and controlled, integrating actuation and sensing capabilities
- Thus, materials that are "smart" and able to adapt to changing conditions

Programmable Matter

- Any type of matter that can algorithmically change its physical properties
  - shape, color, strength, connectivity, stiffness, conductivity, ...
- Change/Transformation is the result of executing an underlying program
  - centralized algorithm, or
  - distributed protocol

Vision:

- Materials that can be programmed and controlled, integrating actuation and sensing capabilities
- Thus, materials that are "smart" and able to adapt to changing conditions

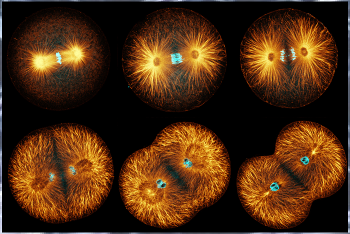
Programmable Matter

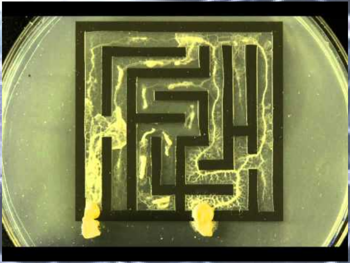
- Any type of matter that can algorithmically change its physical properties
  - shape, color, strength, connectivity, stiffness, conductivity, ...
- Change/Transformation is the result of executing an underlying program
  - centralized algorithm, or
  - distributed protocol

Vision:

- Materials that can be programmed and controlled, integrating actuation and sensing capabilities
- Thus, materials that are "smart" and able to adapt to changing conditions

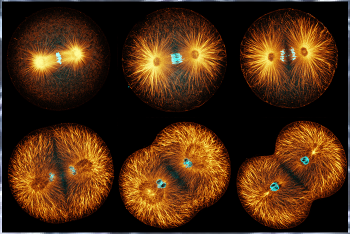
- A wide range of physical/biological systems governed by algorithmic laws
- Usually collections of very large numbers of simple distributed entities
- Higher-level properties are the outcome of coexistence and constant interaction of such entities
- Goals:
  - Reveal the algorithmic aspects of physical systems
  - Develop innovative artificial systems inspired by them

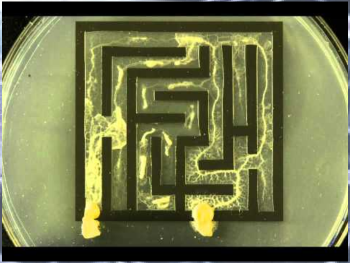




Inspired by Nature

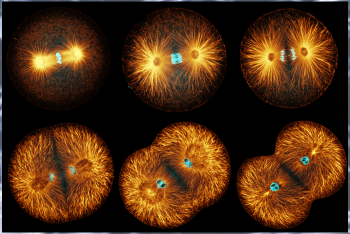
- A wide range of physical/biological systems governed by algorithmic laws
- Usually collections of very large numbers of simple distributed entities
- Higher-level properties are the outcome of coexistence and constant interaction of such entities
- Goals:
  - Reveal the algorithmic aspects of physical systems
  - Develop innovative artificial systems inspired by them

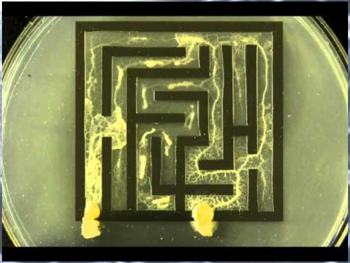




Inspired by Nature

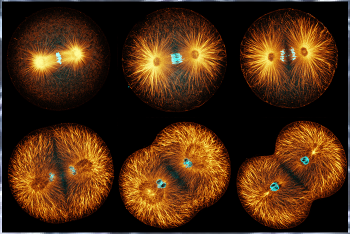
- A wide range of physical/biological systems governed by algorithmic laws
- Usually collections of very large numbers of simple distributed entities
- Higher-level properties are the outcome of coexistence and constant interaction of such entities
- Goals:
  - Reveal the algorithmic aspects of physical systems
  - Develop innovative artificial systems inspired by them

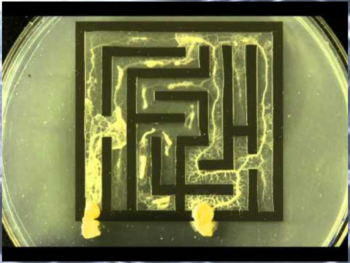




Inspired by Nature

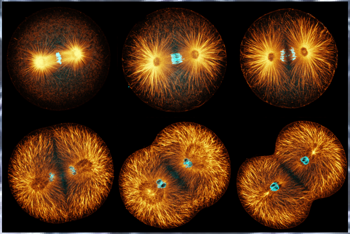
- A wide range of physical/biological systems governed by algorithmic laws
- Usually collections of very large numbers of simple distributed entities
- Higher-level properties are the outcome of coexistence and constant interaction of such entities
- Goals:
  - Reveal the algorithmic aspects of physical systems
  - Develop innovative artificial systems inspired by them

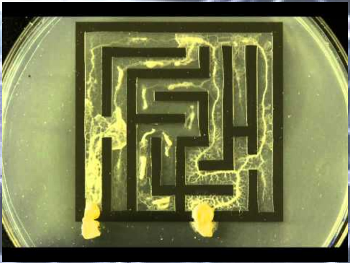




Inspired by Nature

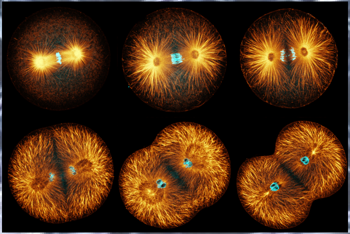
- A wide range of physical/biological systems governed by algorithmic laws
- Usually collections of very large numbers of simple distributed entities
- Higher-level properties are the outcome of coexistence and constant interaction of such entities
- Goals:
  - Reveal the algorithmic aspects of physical systems
  - Develop innovative artificial systems inspired by them

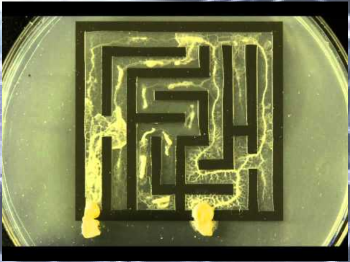




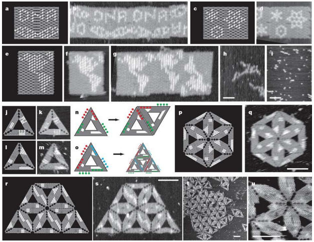
Inspired by Nature

- A wide range of physical/biological systems governed by algorithmic laws
- Usually collections of very large numbers of simple distributed entities
- Higher-level properties are the outcome of coexistence and constant interaction of such entities
- Goals:
  - Reveal the algorithmic aspects of physical systems
  - Develop innovative artificial systems inspired by them





- DNA self-assembly: single-stranded DNA molecules folded into arbitrary nanoscale shapes and patterns [Ro06]



Swarm & Reconfigurable Robotics

- Kilobot [RCN14]: programmable self-assembly of complex 2D shapes by a swarm of 1000 simple autonomous robots
- Robot Pebbles: 1cm cubic modules able to form 2D shapes [GKR10]
- Claytronics [GCM05]: Sub-millimeter, Intel

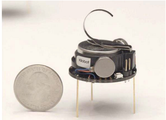
Smart materials

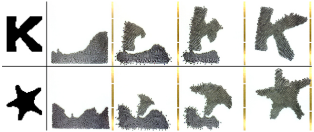
- Optical metamaterials, artificial skins, ...

- DNA self-assembly: single-stranded DNA molecules folded into arbitrary nanoscale shapes and patterns [Ro06]

Swarm & Reconfigurable Robotics

- Kilobot [RCN14]: programmable self-assembly of complex 2D shapes by a swarm of 1000 simple autonomous robots





- Robot Pebbles: 1cm cubic modules able to form 2D shapes [GKR10]
- Claytronics [GCM05]: Sub-millimeter, Intel

Smart materials

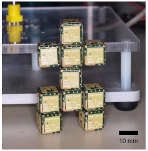
- Optical metamaterials, artificial skins, ...

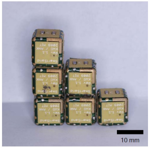
- DNA self-assembly: single-stranded DNA molecules folded into arbitrary nanoscale shapes and patterns [Ro06]

Swarm & Reconfigurable Robotics

- Kilobot [RCN14]: programmable self-assembly of complex 2D shapes by a swarm of 1000 simple autonomous robots

- Robot Pebbles: 1cm cubic modules able to form 2D shapes [GKR10]





- Claytronics [GCM05]: Sub-millimeter, Intel

Smart materials

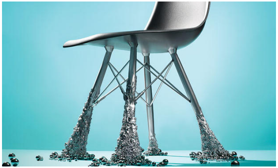
- Optical metamaterials, artificial skins, …

Some Existing Systems

- DNA self-assembly: single-stranded DNA molecules folded into arbitrary nanoscale shapes and patterns [Ro06]

Swarm & Reconfigurable Robotics

- Kilobot [RCN14]: programmable self-assembly of complex 2D shapes by a swarm of 1000 simple autonomous robots
- Robot Pebbles: 1cm cubic modules able to form 2D shapes [GKR10]
- Claytronics [GCM05]: Sub-millimeter, Intel



Smart materials

- Optical metamaterials, artificial skins, ...

Some Existing Systems

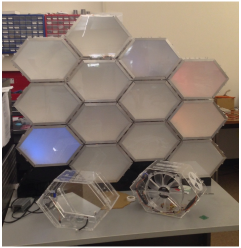
- DNA self-assembly: single-stranded DNA molecules folded into arbitrary nanoscale shapes and patterns [Ro06]

Swarm & Reconfigurable Robotics

- Kilobot [RCN14]: programmable self-assembly of complex 2D shapes by a swarm of 1000 simple autonomous robots
- Robot Pebbles: 1cm cubic modules able to form 2D shapes [GKR10]
- Claytronics [GCM05]: Sub-millimeter, Intel

Smart materials

- Optical metamaterials, artificial skins, ...



- Cellular Automata model neural activity, self-replication, bacterial growth, ...
- Population Protocols [AADFP, PODC '04] and Network Constructors [Michail, Spirakis, PODC '14; Michail, PODC '15]: abstract and simple model of distributed network formation
- Algorithmic self-assembly of DNA: DNA tiles binding to other tiles via Watson-Crick complementary sticky ends
- Models of programmable matter equipped with active mobility/ actuation mechanisms
  - Theories of Mobile, Swarm, and Reconfigurable Robotics
  - Active self-assembly [WCGDWY13]
  - Metamorphic systems [DP04, DSY04]
  - Amoebot model [DDGRSS14, DGSBRS15, DGRSS17]

Some Existing Theoretical Approaches

- Cellular Automata model neural activity, self-replication, bacterial growth, ...
- Population Protocols [AADFP, PODC '04] and Network Constructors [Michail, Spirakis, PODC '14; Michail, PODC '15]: abstract and simple model of distributed network formation
- Algorithmic self-assembly of DNA: DNA tiles binding to other tiles via Watson-Crick complementary sticky ends
- Models of programmable matter equipped with active mobility/ actuation mechanisms
  - Theories of Mobile, Swarm, and Reconfigurable Robotics
  - Active self-assembly [WCGDWY13]
  - Metamorphic systems [DP04, DSY04]
  - Amoebot model [DDGRSS14, DGSBRS15, DGRSS17]

Some Existing Theoretical Approaches

- Cellular Automata model neural activity, self-replication, bacterial growth, ...
- Population Protocols [AADFP, PODC '04] and Network Constructors [Michail, Spirakis, PODC '14; Michail, PODC '15]: abstract and simple model of distributed network formation
- Algorithmic self-assembly of DNA: DNA tiles binding to other tiles via Watson-Crick complementary sticky ends
- Models of programmable matter equipped with active mobility/ actuation mechanisms
  - Theories of Mobile, Swarm, and Reconfigurable Robotics
  - Active self-assembly [WCGDWY13]
  - Metamorphic systems [DP04, DSY04]
  - Amoebot model [DDGRSS14, DGSBRS15, DGRSS17]

Some Existing Theoretical Approaches

- Cellular Automata model neural activity, self-replication, bacterial growth, ...
- Population Protocols [AADFP, PODC '04] and Network Constructors [Michail, Spirakis, PODC '14; Michail, PODC '15]: abstract and simple model of distributed network formation
- Algorithmic self-assembly of DNA: DNA tiles binding to other tiles via Watson-Crick complementary sticky ends
- Models of programmable matter equipped with active mobility/ actuation mechanisms
  - Theories of Mobile, Swarm, and Reconfigurable Robotics
  - Active self-assembly [WCGDWY13]
  - Metamorphic systems [DP04, DSY04]
  - Amoebot model [DDGRSS14, DGSBRS15, DGRSS17]

Some Existing Theoretical Approaches

- Cellular Automata model neural activity, self-replication, bacterial growth, ...
- Population Protocols [AADFP, PODC '04] and Network Constructors [Michail, Spirakis, PODC '14; Michail, PODC '15]: abstract and simple model of distributed network formation
- Algorithmic self-assembly of DNA: DNA tiles binding to other tiles via Watson-Crick complementary sticky ends
- Models of programmable matter equipped with active mobility/ actuation mechanisms
  - Theories of Mobile, Swarm, and Reconfigurable Robotics
  - Active self-assembly [WCGDWY13]
  - Metamorphic systems [DP04, DSY04]
  - Amoebot model [DDGRSS14, DGSBRS15, DGRSS17]

Some Existing Theoretical Approaches

- Cellular Automata model neural activity, self-replication, bacterial growth, ...
- Population Protocols [AADFP, PODC '04] and Network Constructors [Michail, Spirakis, PODC '14; Michail, PODC '15]: abstract and simple model of distributed network formation
- Algorithmic self-assembly of DNA: DNA tiles binding to other tiles via Watson-Crick complementary sticky ends
- Models of programmable matter equipped with active mobility/ actuation mechanisms
  - Theories of Mobile, Swarm, and Reconfigurable Robotics
  - Active self-assembly [WCGDWY13]
  - Metamorphic systems [DP04, DSY04]
  - Amoebot model [DDGRSS14, DGSBRS15, DGRSS17]

Some Existing Theoretical Approaches

- Cellular Automata model neural activity, self-replication, bacterial growth, ...
- Population Protocols [AADFP, PODC '04] and Network Constructors [Michail, Spirakis, PODC '14; Michail, PODC '15]: abstract and simple model of distributed network formation
- Algorithmic self-assembly of DNA: DNA tiles binding to other tiles via Watson-Crick complementary sticky ends
- Models of programmable matter equipped with active mobility/ actuation mechanisms
    - Theories of Mobile, Swarm, and Reconfigurable Robotics
    - Active self-assembly [WCGDWY13]
    - Metamorphic systems [DP04, DSY04]
    - Amoebot model [DDGRSS14, DGSBRS15, DGRSS17]

Some Existing Theoretical Approaches

- Cellular Automata model neural activity, self-replication, bacterial growth, ...
- Population Protocols [AADFP, PODC '04] and Network Constructors [Michail, Spirakis, PODC '14; Michail, PODC '15]: abstract and simple model of distributed network formation
- Algorithmic self-assembly of DNA: DNA tiles binding to other tiles via Watson-Crick complementary sticky ends
- Models of programmable matter equipped with active mobility/ actuation mechanisms
  - Theories of Mobile, Swarm, and Reconfigurable Robotics
  - Active self-assembly [WCGDWY13]
  - Metamorphic systems [DP04, DSY04]
  - Amoebot model [DDGRSS14, DGSBRS15, DGRSS17]

A Motivating System

- A collection of spherical sub-millimeter modules
- Kept together by electrostatic or magnetic forces
- Each module is capable of storing and executing a simple program that
  - handles communication with nearby modules
  - controls the module's capacitors or electromagnets
- Allows a module to rotate or slide over neighboring modules
- The material is able to adjust its shape in a programmable way

A Motivating System

- A collection of spherical sub-millimeter modules
- Kept together by electrostatic or magnetic forces
- Each module is capable of storing and executing a simple program that
  - handles communication with nearby modules
  - controls the module's capacitors or electromagnets
- Allows a module to rotate or slide over neighboring modules
- The material is able to adjust its shape in a programmable way

A Motivating System

- A collection of spherical sub-millimeter modules
- Kept together by electrostatic or magnetic forces
- Each module is capable of storing and executing a simple program that
  - handles communication with nearby modules
  - controls the module's capacitors or electromagnets
- Allows a module to rotate or slide over neighboring modules
- The material is able to adjust its shape in a programmable way

A Motivating System

- A collection of spherical sub-millimeter modules
- Kept together by electrostatic or magnetic forces
- Each module is capable of storing and executing a simple program that
  - handles communication with nearby modules
  - controls the module's capacitors or electromagnets
- Allows a module to rotate or slide over neighboring modules
- The material is able to adjust its shape in a programmable way

A Motivating System

- A collection of spherical sub-millimeter modules
- Kept together by electrostatic or magnetic forces
- Each module is capable of storing and executing a simple program that
  - handles communication with nearby modules
  - controls the module's capacitors or electromagnets
- Allows a module to rotate or slide over neighboring modules
- The material is able to adjust its shape in a programmable way

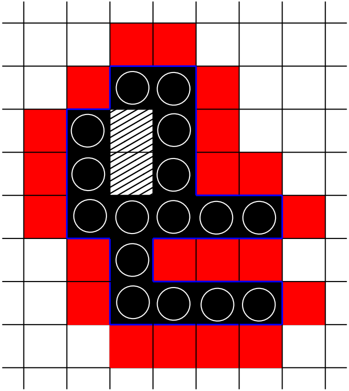
A Motivating System

- A collection of spherical sub-millimeter modules
- Kept together by electrostatic or magnetic forces
- Each module is capable of storing and executing a simple program that
  - handles communication with nearby modules
  - controls the module's capacitors or electromagnets
- Allows a module to rotate or slide over neighboring modules
- The material is able to adjust its shape in a programmable way

A Motivating System

- A collection of spherical sub-millimeter modules
- Kept together by electrostatic or magnetic forces
- Each module is capable of storing and executing a simple program that
  - handles communication with nearby modules
  - controls the module's capacitors or electromagnets
- Allows a module to rotate or slide over neighboring modules
- The material is able to adjust its shape in a programmable way

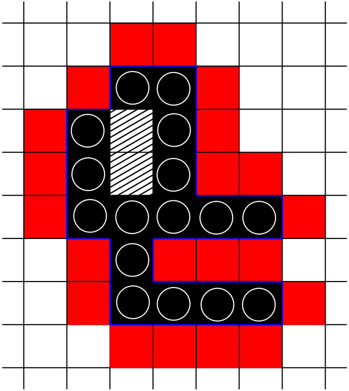
- 2D square grid, (i, j) coordinates
- set V of n modules/nodes
  - spherical, fitting inside a cell of the grid
  - no two nodes may occupy the same cell
- Occupied cells define a shape
  - Connected if horizontal and vertical distance 1 neighborhoods define a connected graph



- Shapes transform to other shapes via a sequence of one or more "valid" movements of individual nodes
- Discrete steps: in every step, 0, 1, or more movements may occur
  - sequential vs parallel case

The Model

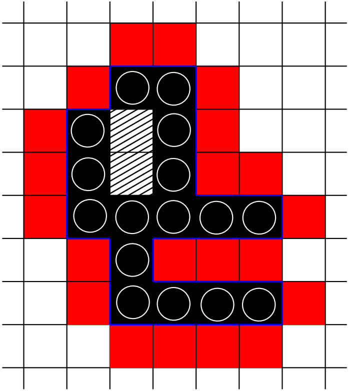
- 2D square grid, (i, j) coordinates
- set V of n modules/nodes
  - spherical, fitting inside a cell of the grid
  - no two nodes may occupy the same cell
- Occupied cells define a shape
  - Connected if horizontal and vertical distance 1 neighborhoods define a connected graph



- Shapes transform to other shapes via a sequence of one or more “valid” movements of individual nodes
- Discrete steps: in every step, 0, 1, or more movements may occur
  - sequential vs parallel case

The Model

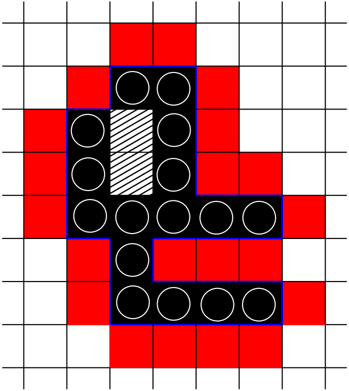
- 2D square grid, (i, j) coordinates
- set V of n modules/nodes
  - spherical, fitting inside a cell of the grid
  - no two nodes may occupy the same cell
- Occupied cells define a shape
  - Connected if horizontal and vertical distance 1 neighborhoods define a connected graph



- Shapes transform to other shapes via a sequence of one or more "valid" movements of individual nodes
- Discrete steps: in every step, 0, 1, or more movements may occur
  - sequential vs parallel case

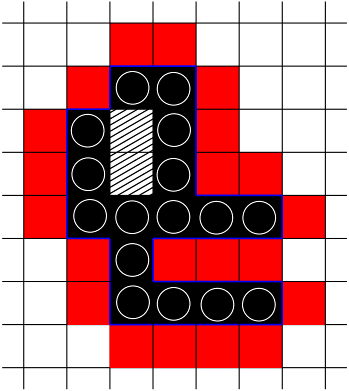
The Model

- 2D square grid, (i, j) coordinates
- set V of n modules/nodes
  - spherical, fitting inside a cell of the grid
  - no two nodes may occupy the same cell
- Occupied cells define a shape
  - Connected if horizontal and vertical distance 1 neighborhoods define a connected graph



- Shapes transform to other shapes via a sequence of one or more "valid" movements of individual nodes
- Discrete steps: in every step, 0, 1, or more movements may occur
  - sequential vs parallel case

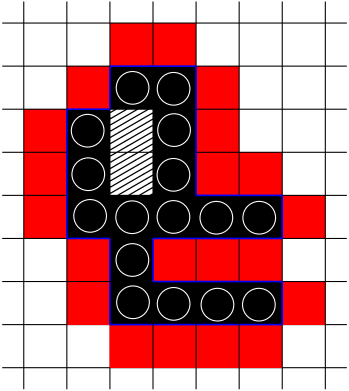
- 2D square grid, (i, j) coordinates
- set V of n modules/nodes
  - spherical, fitting inside a cell of the grid
  - no two nodes may occupy the same cell
- Occupied cells define a shape
  - Connected if horizontal and vertical distance 1 neighborhoods define a connected graph



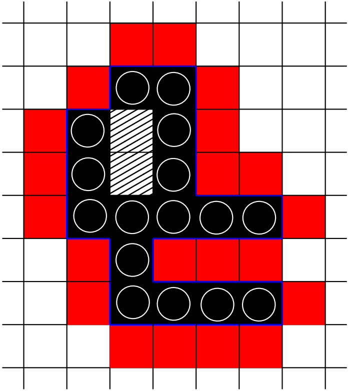
- Shapes transform to other shapes via a sequence of one or more "valid" movements of individual nodes
- Discrete steps: in every step, 0, 1, or more movements may occur
  - sequential vs parallel case

The Model

- 2D square grid, (i, j) coordinates
- set V of n modules/nodes
  - spherical, fitting inside a cell of the grid
  - no two nodes may occupy the same cell
- Occupied cells define a shape
  - Connected if horizontal and vertical distance 1 neighborhoods define a connected graph

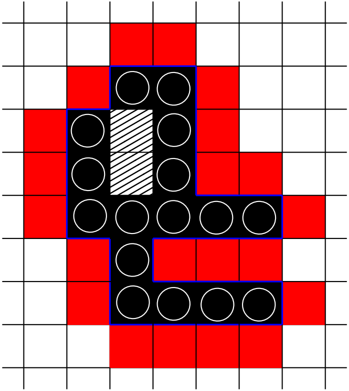


- Shapes transform to other shapes via a sequence of one or more "valid" movements of individual nodes
- Discrete steps: in every step, 0, 1, or more movements may occur
  - sequential vs parallel case



- 2D square grid, (i, j) coordinates
- set V of n modules/nodes
  - spherical, fitting inside a cell of the grid
  - no two nodes may occupy the same cell
- Occupied cells define a shape
  - Connected if horizontal and vertical distance 1 neighborhoods define a connected graph
- Shapes transform to other shapes via a sequence of one or more "valid" movements of individual nodes
- Discrete steps: in every step, 0, 1, or more movements may occur
  - sequential vs parallel case

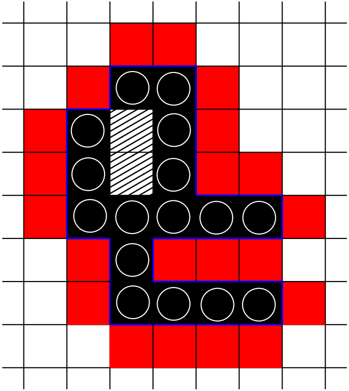
- 2D square grid, (i, j) coordinates
- set V of n modules/nodes
  - spherical, fitting inside a cell of the grid
  - no two nodes may occupy the same cell
- Occupied cells define a shape
  - Connected if horizontal and vertical distance 1 neighborhoods define a connected graph



- Shapes transform to other shapes via a sequence of one or more "valid" movements of individual nodes
- Discrete steps: in every step, 0, 1, or more movements may occur
  - sequential vs parallel case

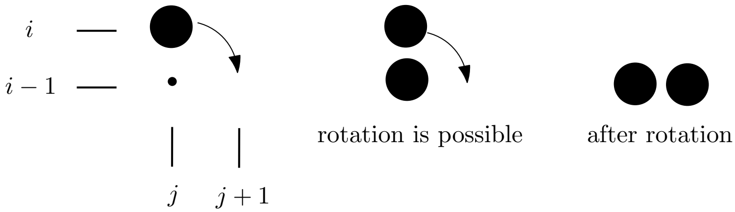
The Model

- 2D square grid, (i, j) coordinates
- set V of n modules/nodes
  - spherical, fitting inside a cell of the grid
  - no two nodes may occupy the same cell
- Occupied cells define a shape
  - Connected if horizontal and vertical distance 1 neighborhoods define a connected graph



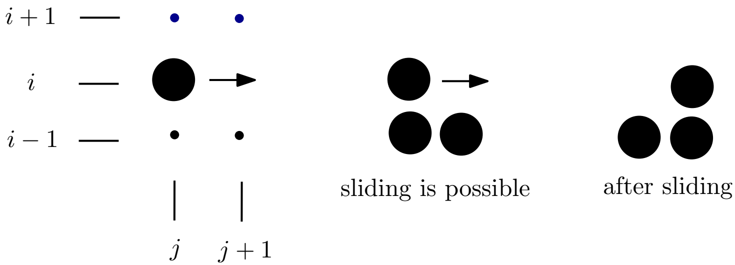
- Shapes transform to other shapes via a sequence of one or more "valid" movements of individual nodes
- Discrete steps: in every step, 0, 1, or more movements may occur
  - sequential vs parallel case

Rotation



Simple and Feasible Actuation Mechanisms

Sliding



Main Question

- *Can an initial arrangement of the material transform to some other target arrangement?*
  - Either by an external authority or by itself

In more technical terms:

- We are provided with an initial shape A and a target shape B and we are asked whether A can be transformed to B via a sequence of valid transformation steps
  - If yes, give also such a sequence (optimizing some parameters)
- Steps are
  - rotations only
  - rotations and slidings
- Possibly additional constraints: connectivity preservation, restricted area, labeled nodes, ...

Main Question

- *Can an initial arrangement of the material transform to some other target arrangement?*
  - Either by an external authority or by itself

In more technical terms:

- We are provided with an initial shape A and a target shape B and we are asked whether A can be transformed to B via a sequence of valid transformation steps
  - If yes, give also such a sequence (optimizing some parameters)
- Steps are
  - rotations only
  - rotations and slidings
- Possibly additional constraints: connectivity preservation, restricted area, labeled nodes, ...

Main Question

- *Can an initial arrangement of the material transform to some other target arrangement?*
  - Either by an external authority or by itself

In more technical terms:

- We are provided with an initial shape A and a target shape B and we are asked whether A can be transformed to B via a sequence of valid transformation steps
  - If yes, give also such a sequence (optimizing some parameters)
- Steps are
  - rotations only
  - rotations and slidings
- Possibly additional constraints: connectivity preservation, restricted area, labeled nodes, . . .

Main Question

- *Can an initial arrangement of the material transform to some other target arrangement?*
  - Either by an external authority or by itself

In more technical terms:

- We are provided with an initial shape A and a target shape B and we are asked whether A can be transformed to B via a sequence of valid transformation steps
  - If yes, give also such a sequence (optimizing some parameters)
- Steps are
  - rotations only
  - rotations and slidings
- Possibly additional constraints: connectivity preservation, restricted area, labeled nodes, ...

Main Question

- *Can an initial arrangement of the material transform to some other target arrangement?*
  - Either by an external authority or by itself

In more technical terms:

- We are provided with an initial shape A and a target shape B and we are asked whether A can be transformed to B via a sequence of valid transformation steps
  - If yes, give also such a sequence (optimizing some parameters)
- Steps are
  - rotations only
  - rotations and slidings
- Possibly additional constraints: connectivity preservation, restricted area, labeled nodes, ...

- *Can an initial arrangement of the material transform to some other target arrangement?*
  - Either by an external authority or by itself

In more technical terms:

- We are provided with an initial shape A and a target shape B and we are asked whether A can be transformed to B via a sequence of valid transformation steps
  - If yes, give also such a sequence (optimizing some parameters)
- Steps are
  - rotations only
  - rotations and slidings
- Possibly additional constraints: connectivity preservation, restricted area, labeled nodes, . . .

Main Question

- *Can an initial arrangement of the material transform to some other target arrangement?*
  - Either by an external authority or by itself

In more technical terms:

- We are provided with an initial shape A and a target shape B and we are asked whether A can be transformed to B via a sequence of valid transformation steps
  - If yes, give also such a sequence (optimizing some parameters)
- Steps are
  - rotations only
  - rotations and slidings
- Possibly additional constraints: connectivity preservation, restricted area, labeled nodes, ...

Main Question

- *Can an initial arrangement of the material transform to some other target arrangement?*
  - Either by an external authority or by itself

In more technical terms:

- We are provided with an initial shape A and a target shape B and we are asked whether A can be transformed to B via a sequence of valid transformation steps
  - If yes, give also such a sequence (optimizing some parameters)
- Steps are
  - rotations only
  - rotations and slidings
- Possibly additional constraints: connectivity preservation, restricted area, labeled nodes, ...

Main Question

- *Can an initial arrangement of the material transform to some other target arrangement?*
  - Either by an external authority or by itself

In more technical terms:

- We are provided with an initial shape A and a target shape B and we are asked whether A can be transformed to B via a sequence of valid transformation steps
  - If yes, give also such a sequence (optimizing some parameters)
- Steps are
  - rotations only
  - rotations and slidings
- Possibly additional constraints: connectivity preservation, restricted area, labeled nodes, . . .

Rotation Only

Problem

ROT-TRANSFORMABILITY. *Given an initial shape A and a target shape B (usually both connected), decide whether A can be transformed to B by a sequence of rotation only movements.*

- Is there (in principle) a sequence of rotation movements that achieves the transformation?
- Main theorem: ROT-TRANSFORMABILITY $\in$ **P**

Rotation Only

Problem

ROT-TRANSFORMABILITY. *Given an initial shape A and a target shape B (usually both connected), decide whether A can be transformed to B by a sequence of rotation only movements.*

- Is there (in principle) a sequence of rotation movements that achieves the transformation?
- Main theorem: ROT-TRANSFORMABILITY $\in$ **P**

Rotation Only

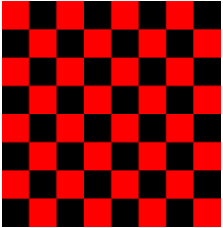
Problem

ROT-TRANSFORMABILITY. *Given an initial shape A and a target shape B (usually both connected), decide whether A can be transformed to B by a sequence of rotation only movements.*

- Is there (in principle) a sequence of rotation movements that achieves the transformation?
- Main theorem: ROT-TRANSFORMABILITY $\in$ **P**

A Lower Bound

- A **black**-**red** checkered coloring of the 2D grid
- Each node of the material occupies at any given time a distinct cell



Observation

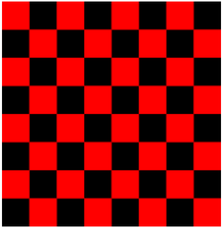
Rotation is color-preserving. Formally, $A \overset{r}{\rightsquigarrow} B \Rightarrow A$ and B are color-consistent.

- *Every node beginning from a* **black** *(**red**) position of the grid, will always be on* **black** *(**red**, respectively) positions throughout the transformation*

$\Rightarrow$ Color-inconsistent shapes cannot be transformed to each other

A Lower Bound

- A **black**-**red** checkered coloring of the 2D grid
- Each node of the material occupies at any given time a distinct cell



Observation

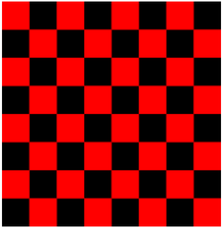
Rotation is color-preserving. Formally, $A \overset{r}{\rightsquigarrow} B \Rightarrow A$ and B are color-consistent.

- *Every node beginning from a* **black** *(**red**) position of the grid, will always be on* **black** *(**red**, respectively) positions throughout the transformation*

$\Rightarrow$ Color-inconsistent shapes cannot be transformed to each other

A Lower Bound

- A **black**-**red** checkered coloring of the 2D grid
- Each node of the material occupies at any given time a distinct cell



Observation

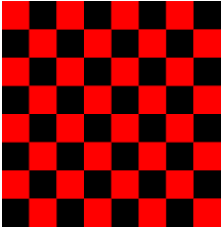
Rotation is color-preserving. Formally, $A \stackrel{r}{\rightsquigarrow} B \Rightarrow A$ and B are color-consistent.

- *Every node beginning from a* **black** *(***red***) position of the grid, will always be on* **black** *(***red***, respectively) positions throughout the transformation*

$\Rightarrow$ Color-inconsistent shapes cannot be transformed to each other

A Lower Bound

- A **black**-**red** checkered coloring of the 2D grid
- Each node of the material occupies at any given time a distinct cell



Observation

Rotation is color-preserving. Formally, $A \overset{r}{\rightsquigarrow} B \Rightarrow A$ and B are color-consistent.

- *Every node beginning from a* **black** (**red**) *position of the grid, will always be on* **black** (**red**, *respectively) positions throughout the transformation*

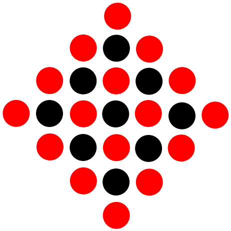
$\Rightarrow$ Color-inconsistent shapes cannot be transformed to each other

A Blocked Shape

- But color-consistency does not necessarily imply feasible transformation
- A rhombus cannot move at all
- Still it has a color-consistent version from the line-with-leaves family
- Any connected shape has a color-consistent version from the line-with-leaves family

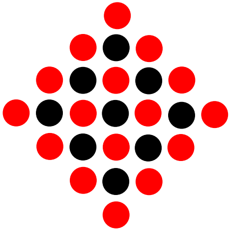
A Blocked Shape

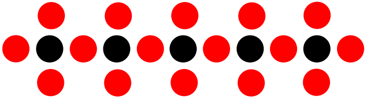
- But color-consistency does not necessarily imply feasible transformation
- A rhombus cannot move at all
- Still it has a color-consistent version from the line-with-leaves family
- Any connected shape has a color-consistent version from the line-with-leaves family



A Blocked Shape

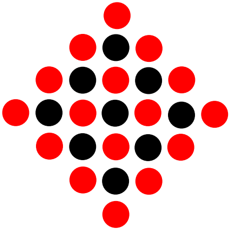
- But color-consistency does not necessarily imply feasible transformation
- A rhombus cannot move at all
- Still it has a color-consistent version from the line-with-leaves family
- Any connected shape has a color-consistent version from the line-with-leaves family

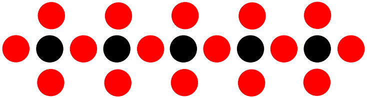




A Blocked Shape

- But color-consistency does not necessarily imply feasible transformation
- A rhombus cannot move at all
- Still it has a color-consistent version from the line-with-leaves family
- Any connected shape has a color-consistent version from the line-with-leaves family





Rotation Only: Main Theorem

Theorem

Rot-Transformability $\in$ **P**.

- Independent of the location and number of these initial movements

Rotation Only: Main Theorem

Theorem

ROT-TRANSFORMABILITY $\in$ **P**.

Proof

- *If A and B are color-inconsistent*
  - *Reject*
- *Otherwise, check whether each has an available movement*
  - *If at least one is blocked*
    - *Reject*
  - *Otherwise (i.e., both can move)*
    - *Accept*

- Independent of the location and number of these initial movements

Rotation Only: Main Theorem

Theorem

ROT-TRANSFORMABILITY $\in$ **P**.

Proof

- *If A and B are color-inconsistent*
  - *Reject*
- *Otherwise, check whether each has an available movement*
  - *If at least one is blocked*
    - *Reject*
  - *Otherwise (i.e., both can move)*
    - *Accept*

- Independent of the location and number of these initial movements

Rotation Only: Main Theorem

Theorem

ROT-TRANSFORMABILITY $\in$ **P**.

Proof

- *If A and B are color-inconsistent*
  - *Reject*
- *Otherwise, check whether each has an available movement*
  - *If at least one is blocked*
    - *Reject*
  - *Otherwise (i.e., both can move)*
    - *Accept*

- Independent of the location and number of these initial movements

Rotation Only: Main Theorem

Theorem

ROT-TRANSFORMABILITY $\in$ **P**.

Proof

- *If A and B are color-inconsistent*
  - *Reject*
- *Otherwise, check whether each has an available movement*
  - *If at least one is blocked*
    - *Reject*
  - *Otherwise (i.e., both can move)*
    - *Accept*

- Independent of the location and number of these initial movements

Rotation Only: Main Theorem

Theorem

ROT-TRANSFORMABILITY $\in$ **P**.

Proof

- *If A and B are color-inconsistent*
  - *Reject*
- *Otherwise, check whether each has an available movement*
  - *If at least one is blocked*
    - *Reject*
  - *Otherwise (i.e., both can move)*
    - *Accept*

- Independent of the location and number of these initial movements

Rotation Only: Main Theorem

Theorem

ROT-TRANSFORMABILITY $\in$ **P**.

Proof

- *If A and B are color-inconsistent*
  - *Reject*
- *Otherwise, check whether each has an available movement*
  - *If at least one is blocked*
    - *Reject*
  - *Otherwise (i.e., both can move)*
    - *Accept*

- Independent of the location and number of these initial movements

Rotation Only: Main Theorem

Theorem

ROT-TRANSFORMABILITY $\in$ **P**.

Proof

- *If A and B are color-inconsistent*
  - *Reject*
- *Otherwise, check whether each has an available movement*
  - *If at least one is blocked*
    - *Reject*
  - *Otherwise (i.e., both can move)*
    - *Accept*

- Independent of the location and number of these initial movements

Rotation Only: Main Theorem

Theorem

ROT-TRANSFORMABILITY $\in$ **P**.

Proof

- *If A and B are color-inconsistent*
  - *Reject*
- *Otherwise, check whether each has an available movement*
  - *If at least one is blocked*
    - *Reject*
  - *Otherwise (i.e., both can move)*
    - *Accept*

- Independent of the location and number of these initial movements

Rotation Only: Main Theorem

- *Given any connected shape C and a 2-seed outside it, we can transform C to its color-consistent line-with-leaves*
- *A and B are color-consistent so they can be transformed to the same line-with-leaves L*
- *A can be transformed to L and by reversibility of rotation L to B, therefore, A to B*
- *If the available movement is on the perimeter, then immediate to extract the required 2-seed*
- *If it is in the interior, we can still prove that it can be propagated to the perimeter*

Rotation Only: Main Theorem

- *Given any connected shape C and a 2-seed outside it, we can transform C to its color-consistent line-with-leaves*
- *A and B are color-consistent so they can be transformed to the same line-with-leaves L*
- *A can be transformed to L and by reversibility of rotation L to B, therefore, A to B*
- *If the available movement is on the perimeter, then immediate to extract the required 2-seed*
- *If it is in the interior, we can still prove that it can be propagated to the perimeter*

□

Rotation Only: Main Theorem

- *Given any connected shape C and a 2-seed outside it, we can transform C to its color-consistent line-with-leaves*
- *A and B are color-consistent so they can be transformed to the same line-with-leaves L*
- *A can be transformed to L and by reversibility of rotation L to B, therefore, A to B*
- *If the available movement is on the perimeter, then immediate to extract the required 2-seed*
- *If it is in the interior, we can still prove that it can be propagated to the perimeter*

$\square$

Rotation Only: Main Theorem

- *Given any connected shape C and a 2-seed outside it, we can transform C to its color-consistent line-with-leaves*
- *A and B are color-consistent so they can be transformed to the same line-with-leaves L*
- *A can be transformed to L and by reversibility of rotation L to B, therefore, A to B*
- *If the available movement is on the perimeter, then immediate to extract the required 2-seed*
- *If it is in the interior, we can still prove that it can be propagated to the perimeter*

□

Rotation Only: Main Theorem

- *Given any connected shape C and a 2-seed outside it, we can transform C to its color-consistent line-with-leaves*
- *A and B are color-consistent so they can be transformed to the same line-with-leaves L*
- *A can be transformed to L and by reversibility of rotation L to B, therefore, A to B*
- *If the available movement is on the perimeter, then immediate to extract the required 2-seed*
- *If it is in the interior, we can still prove that it can be propagated to the perimeter*

$\square$

Rotation Only + Connectivity Preservation

Problem

ROTC-TRANSFORMABILITY. *Given an initial shape A and a target shape B, both connected, decide whether A can be transformed to B by a sequence of rotation only movements if connectivity must be preserved throughout the transformation.*

- guarantees communication maintained
- minimizes transformation failures
- requires less sophisticated actuation mechanisms
- increases external forces required to break the system apart

Rotation Only + Connectivity Preservation

Problem

ROTC-TRANSFORMABILITY. *Given an initial shape A and a target shape B, both connected, decide whether A can be transformed to B by a sequence of rotation only movements if connectivity must be preserved throughout the transformation.*

Why connectivity preservation?

- guarantees communication maintained
- minimizes transformation failures
- requires less sophisticated actuation mechanisms
- increases external forces required to break the system apart

Rotation Only + Connectivity Preservation

Problem

ROTC-TRANSFORMABILITY. *Given an initial shape A and a target shape B, both connected, decide whether A can be transformed to B by a sequence of rotation only movements if connectivity must be preserved throughout the transformation.*

Why connectivity preservation?

- guarantees communication maintained
- minimizes transformation failures
- requires less sophisticated actuation mechanisms
- increases external forces required to break the system apart

Rotation Only + Connectivity Preservation

Problem

ROTC-TRANSFORMABILITY. *Given an initial shape A and a target shape B, both connected, decide whether A can be transformed to B by a sequence of rotation only movements if connectivity must be preserved throughout the transformation.*

Why connectivity preservation?

- guarantees communication maintained
- minimizes transformation failures
- requires less sophisticated actuation mechanisms
- increases external forces required to break the system apart

Rotation Only + Connectivity Preservation

Problem

ROTC-TRANSFORMABILITY. *Given an initial shape A and a target shape B, both connected, decide whether A can be transformed to B by a sequence of rotation only movements if connectivity must be preserved throughout the transformation.*

Why connectivity preservation?

- guarantees communication maintained
- minimizes transformation failures
- requires less sophisticated actuation mechanisms
- increases external forces required to break the system apart

Rotation Only + Connectivity Preservation

Problem

ROTC-TRANSFORMABILITY. *Given an initial shape A and a target shape B, both connected, decide whether A can be transformed to B by a sequence of rotation only movements if connectivity must be preserved throughout the transformation.*

Why connectivity preservation?

- guarantees communication maintained
- minimizes transformation failures
- requires less sophisticated actuation mechanisms
- increases external forces required to break the system apart

Theorem

ROTC-TRANSFORMABILITY $\in$ **PSPACE**.

RotC-Transformability Inclusions

Theorem

RotC-Transformability $\in$ **PSPACE**.

Theorem

RotC-Transformability $\subset$ Rot-Transformability.

RotC-Transformability Inclusions

Theorem

RotC-Transformability $\in$ **PSPACE**.

Theorem

RotC-Transformability $\subset$ Rot-Transformability.

Proof

- *Suffices to give a transformation problem in* Rot-Transformability$\setminus$RotC-Transformability
- *Line folding:* $\notin$ RotC-Transformability *for any* $n > 4$
  - *Trapped in endpoints rotation loop*

RotC-Transformability Inclusions

Theorem

RotC-Transformability $\in$ **PSPACE**.

Theorem

RotC-Transformability $\subset$ Rot-Transformability.

Proof

- *Suffices to give a transformation problem in* Rot-Transformability\RotC-Transformability
- *Line folding:* $\notin$ RotC-Transformability *for any* $n > 4$
  - *Trapped in endpoints rotation loop*

RotC-Transformability Inclusions

Theorem

RotC-Transformability $\in$ **PSPACE**.

Theorem

RotC-Transformability $\subset$ Rot-Transformability.

Proof

- *Suffices to give a transformation problem in* Rot-Transformability $\setminus$ RotC-Transformability
- *Line folding:* $\notin$ RotC-Transformability *for any* $n > 4$
    - *Trapped in endpoints rotation loop*

RotC-Transformability Inclusions

Theorem

RotC-Transformability $\in$ **PSPACE**.

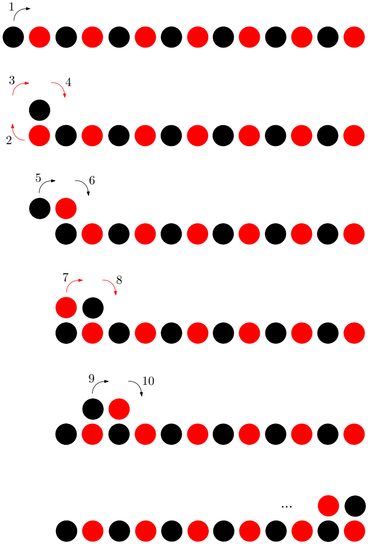
Theorem

RotC-Transformability $\subset$ Rot-Transformability.

Proof

- *Suffices to give a transformation problem in* Rot-Transformability$\setminus$RotC-Transformability
- *Line folding:* $\notin$ RotC-Transformability *for any* $n > 4$
  - *Trapped in endpoints rotation loop*

Line folding $\in$ Rot-Transformability



$\square$

- A small auxiliary shape attached to the initial shape
- Can make feasible otherwise infeasible transformations
    - Some existing systems make use of seeds
- 3-line seed, if placed appropriately, enables folding of a line with connectivity preservation

Seeds

- A small auxiliary shape attached to the initial shape
- Can make feasible otherwise infeasible transformations
  - Some existing systems make use of seeds
- 3-line seed, if placed appropriately, enables folding of a line with connectivity preservation

Seeds

- A small auxiliary shape attached to the initial shape
- Can make feasible otherwise infeasible transformations
  - Some existing systems make use of seeds
- 3-line seed, if placed appropriately, enables folding of a line with connectivity preservation

Seeds

- A small auxiliary shape attached to the initial shape
- Can make feasible otherwise infeasible transformations
  - Some existing systems make use of seeds

Problem

Minimum-Seed-Determination. Determine a minimum-size seed and an initial positioning of that seed relative to A that makes the transformation from A to B feasible.

- 3-line seed, if placed appropriately, enables folding of a line with connectivity preservation

Seeds

- A small auxiliary shape attached to the initial shape
- Can make feasible otherwise infeasible transformations
  - Some existing systems make use of seeds

Problem

Minimum-Seed-Determination. Determine a minimum-size seed and an initial positioning of that seed relative to A that makes the transformation from A to B feasible.

Example:

- 3-line seed, if placed appropriately, enables folding of a line with connectivity preservation

Seeds

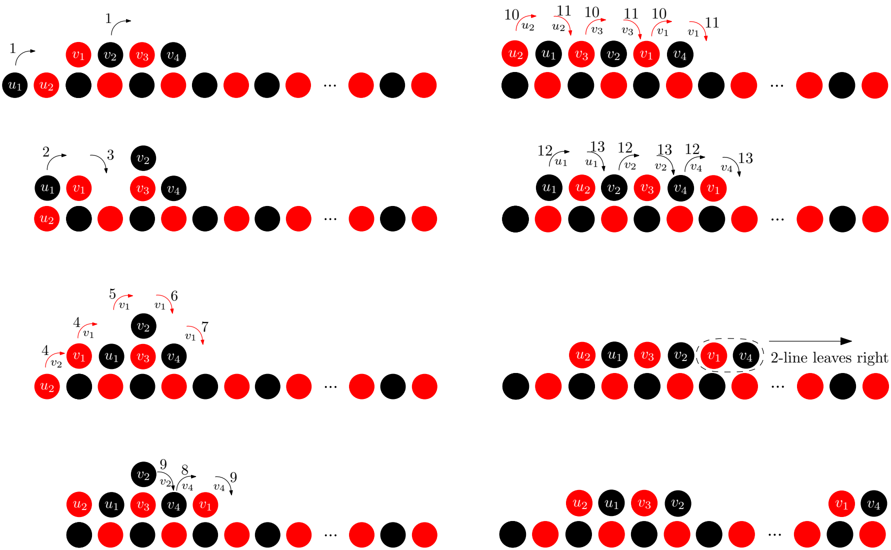
- A small auxiliary shape attached to the initial shape
- Can make feasible otherwise infeasible transformations
  - Some existing systems make use of seeds

Problem

Minimum-Seed-Determination. Determine a minimum-size seed and an initial positioning of that seed relative to A that makes the transformation from A to B feasible.

Example:

- 3-line seed, if placed appropriately, enables folding of a line with connectivity preservation



2-line leaves right

- Minimum seed that can walk the perimeter of any shape
- Could be able to move nodes gradually to a predetermined position, in order to transform the initial shape into a line-with-leaves
  - without ever breaking connectivity
- Also an attempt to simulate the universal transformations based on combined rotation and sliding (discussed later on)

- Open problems: When can such seeds be extracted? Can all nodes be transferred?

- Minimum seed that can walk the perimeter of any shape
- Could be able to move nodes gradually to a predetermined position, in order to transform the initial shape into a line-with-leaves
  - without ever breaking connectivity
- Also an attempt to simulate the universal transformations based on combined rotation and sliding (discussed later on)

- Open problems: When can such seeds be extracted? Can all nodes be transferred?

Seeds and ROTC-TRANSFORMABILITY

- Minimum seed that can walk the perimeter of any shape
- Could be able to move nodes gradually to a predetermined position, in order to transform the initial shape into a line-with-leaves
  - without ever breaking connectivity
- Also an attempt to simulate the universal transformations based on combined rotation and sliding (discussed later on)

- Open problems: When can such seeds be extracted? Can all nodes be transferred?

Seeds and RotC-Transformability

- Minimum seed that can walk the perimeter of any shape
- Could be able to move nodes gradually to a predetermined position, in order to transform the initial shape into a line-with-leaves
  - without ever breaking connectivity
- Also an attempt to simulate the universal transformations based on combined rotation and sliding (discussed later on)

- Open problems: When can such seeds be extracted? Can all nodes be transferred?

Seeds and RotC-Transformability

- Minimum seed that can walk the perimeter of any shape
- Could be able to move nodes gradually to a predetermined position, in order to transform the initial shape into a line-with-leaves
  - without ever breaking connectivity
- Also an attempt to simulate the universal transformations based on combined rotation and sliding (discussed later on)

Theorem

If connectivity must be preserved: (i) Any (≤ 4)-seed cannot traverse the perimeter of a line, (ii) A 6-seed can traverse the perimeter of any discrete-convex shape.

- Open problems: When can such seeds be extracted? Can all nodes be transferred?

Seeds and RotC-Transformability

- Minimum seed that can walk the perimeter of any shape
- Could be able to move nodes gradually to a predetermined position, in order to transform the initial shape into a line-with-leaves
  - without ever breaking connectivity
- Also an attempt to simulate the universal transformations based on combined rotation and sliding (discussed later on)

Theorem

If connectivity must be preserved: (i) Any (≤ 4)-seed cannot traverse the perimeter of a line, (ii) A 6-seed can traverse the perimeter of any discrete-convex shape.

- Open problems: When can such seeds be extracted? Can all nodes be transferred?

Rotation + Sliding

Problem

RS-TRANSFORMABILITY. *Given an initial shape A and a target shape B (usually both connected), decide whether A can be transformed to B by a sequence of rotation and sliding movements.*

- Universality can be proved [DP04]: Any A can be transformed to any B of the same size, without ever breaking the connectivity during the transformation

Rotation + Sliding

Problem

RS-TRANSFORMABILITY. *Given an initial shape A and a target shape B (usually both connected), decide whether A can be transformed to B by a sequence of rotation and sliding movements.*

- Universality can be proved [DP04]: Any A can be transformed to any B of the same size, without ever breaking the connectivity during the transformation

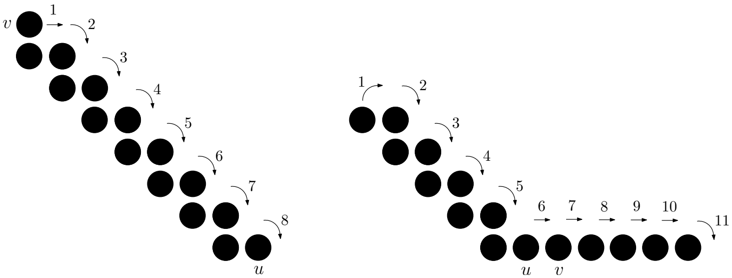
Time Analysis

- The universal transformation requires $\Theta(n^2)$ movements in the worst case
- There are pairs of shapes for which any strategy may require a quadratic number of steps to transform one shape to the other
    - A ladder to a line
    - Potential-function argument based on distance

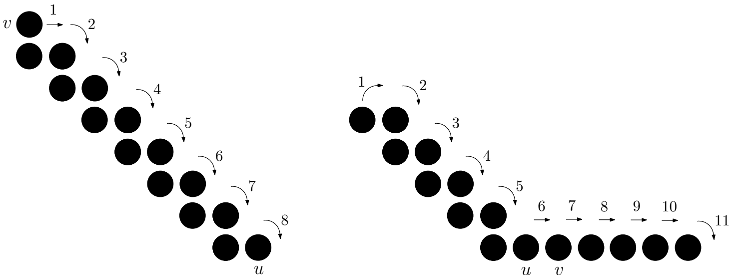
Time Analysis

- The universal transformation requires $\Theta(n^2)$ movements in the worst case
- There are pairs of shapes for which any strategy may require a quadratic number of steps to transform one shape to the other
  - A ladder to a line
  - Potential-function argument based on distance

- The universal transformation requires $\Theta(n^2)$ movements in the worst case
- There are pairs of shapes for which any strategy may require a quadratic number of steps to transform one shape to the other
  - A ladder to a line
  - Potential-function argument based on distance



- The universal transformation requires $\Theta(n^2)$ movements in the worst case
- There are pairs of shapes for which any strategy may require a quadratic number of steps to transform one shape to the other
  - A ladder to a line
  - Potential-function argument based on distance



Pipelining

Theorem

Let A and B be any connected shapes, such that $|A| = |B| = n$. Then there is a pipelining strategy that can transform A to B (and inversely) by rotations and slidings, without breaking the connectivity during the transformation, in $O(n)$ parallel time. Even in the node-labeled version of the problem.

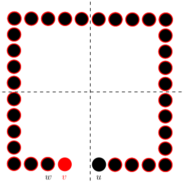
Pipelining

Theorem

Let A and B be any connected shapes, such that $|A| = |B| = n$. Then there is a pipelining strategy that can transform A to B (and inversely) by rotations and slidings, without breaking the connectivity during the transformation, in $O(n)$ parallel time. Even in the node-labeled version of the problem.

Proposition

There are two shapes A and B with $d(A, B) = 2$, such that A and B require $\Omega(n)$ parallel time to be transformed to each other.



Open Problems

- Allow some restricted degree of connectivity breaking
- Enrich with physical properties: strength, mass balancing, energy balancing, statistical failures
- Restrict the maximum allowed area or dimensions of a transformation
  - Leads to models equivalent to several interesting puzzles [De01]
- More sophisticated mechanical operations
  - Would enable a larger set of transformations and reduce time complexity
  - e.g., parallel insertion/extraction, move whole subshapes

Open Problems

- Allow some restricted degree of connectivity breaking
- Enrich with physical properties: strength, mass balancing, energy balancing, statistical failures
- Restrict the maximum allowed area or dimensions of a transformation
  - Leads to models equivalent to several interesting puzzles [De01]
- More sophisticated mechanical operations
  - Would enable a larger set of transformations and reduce time complexity
  - e.g., parallel insertion/extraction, move whole subshapes

Open Problems

- Allow some restricted degree of connectivity breaking
- Enrich with physical properties: strength, mass balancing, energy balancing, statistical failures
- Restrict the maximum allowed area or dimensions of a transformation
  - Leads to models equivalent to several interesting puzzles [De01]
- More sophisticated mechanical operations
  - Would enable a larger set of transformations and reduce time complexity
  - e.g., parallel insertion/extraction, move whole subshapes

Open Problems

- Allow some restricted degree of connectivity breaking
- Enrich with physical properties: strength, mass balancing, energy balancing, statistical failures
- Restrict the maximum allowed area or dimensions of a transformation
  - Leads to models equivalent to several interesting puzzles [De01]
- More sophisticated mechanical operations
  - Would enable a larger set of transformations and reduce time complexity
  - e.g., parallel insertion/extraction, move whole subshapes

Open Problems

- Allow some restricted degree of connectivity breaking
- Enrich with physical properties: strength, mass balancing, energy balancing, statistical failures
- Restrict the maximum allowed area or dimensions of a transformation
  - Leads to models equivalent to several interesting puzzles [De01]
- More sophisticated mechanical operations
  - Would enable a larger set of transformations and reduce time complexity
  - e.g., parallel insertion/extraction, move whole subshapes

Open Problems

- Allow some restricted degree of connectivity breaking
- Enrich with physical properties: strength, mass balancing, energy balancing, statistical failures
- Restrict the maximum allowed area or dimensions of a transformation
  - Leads to models equivalent to several interesting puzzles [De01]
- More sophisticated mechanical operations
  - Would enable a larger set of transformations and reduce time complexity
  - e.g., parallel insertion/extraction, move whole subshapes

Open Problems

- Allow some restricted degree of connectivity breaking
- Enrich with physical properties: strength, mass balancing, energy balancing, statistical failures
- Restrict the maximum allowed area or dimensions of a transformation
  - Leads to models equivalent to several interesting puzzles [De01]
- More sophisticated mechanical operations
  - Would enable a larger set of transformations and reduce time complexity
  - e.g., parallel insertion/extraction, move whole subshapes

Open Problems

- What is the exact complexity of ROTC-TRANSFORMABILITY?
- What is the complexity of computing the optimum transformation? Can it be satisfactorily approximated?
- Consider other module-shapes, e.g., hexagons
  - Geometry changes, different actuation possibilities, not directly comparable to disks
- 3D transformations and labeled transformations
- Higher level properties: global functionality, response to stimuli, self-repair
- Distributed transformations

Open Problems

- What is the exact complexity of RotC-Transformability?
- What is the complexity of computing the optimum transformation? Can it be satisfactorily approximated?
- Consider other module-shapes, e.g., hexagons
  - Geometry changes, different actuation possibilities, not directly comparable to disks
- 3D transformations and labeled transformations
- Higher level properties: global functionality, response to stimuli, self-repair
- Distributed transformations

Open Problems

- What is the exact complexity of RotC-Transformability?
- What is the complexity of computing the optimum transformation? Can it be satisfactorily approximated?
- Consider other module-shapes, e.g., hexagons
  - Geometry changes, different actuation possibilities, not directly comparable to disks
- 3D transformations and labeled transformations
- Higher level properties: global functionality, response to stimuli, self-repair
- Distributed transformations

Open Problems

- What is the exact complexity of RotC-Transformability?
- What is the complexity of computing the optimum transformation? Can it be satisfactorily approximated?
- Consider other module-shapes, e.g., hexagons
  - Geometry changes, different actuation possibilities, not directly comparable to disks
- 3D transformations and labeled transformations
- Higher level properties: global functionality, response to stimuli, self-repair
- Distributed transformations

Open Problems

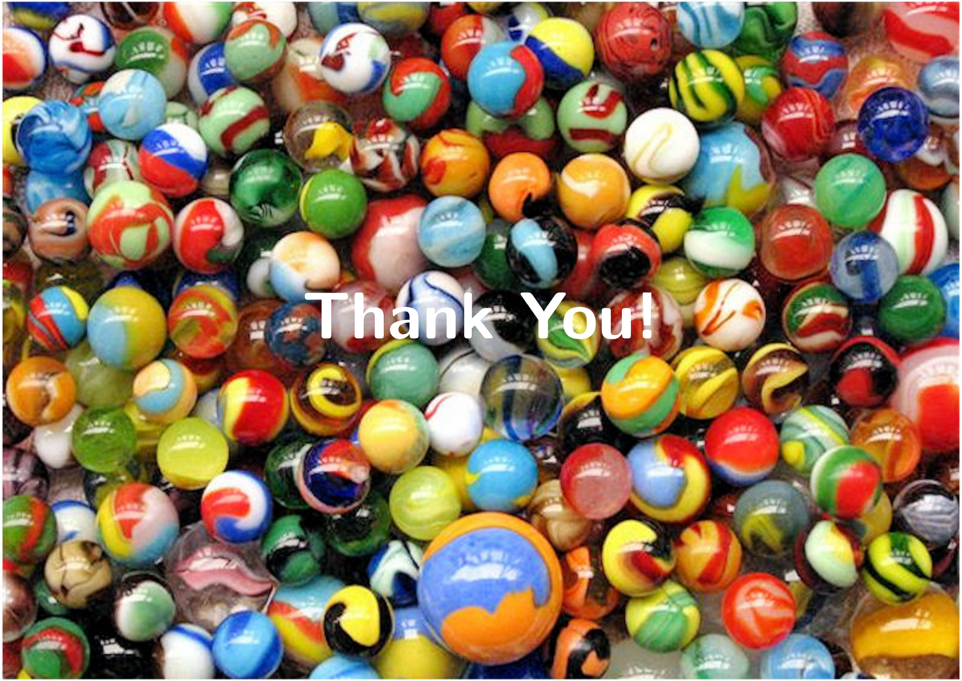
- What is the exact complexity of ROTC-TRANSFORMABILITY?
- What is the complexity of computing the optimum transformation? Can it be satisfactorily approximated?
- Consider other module-shapes, e.g., hexagons
  - Geometry changes, different actuation possibilities, not directly comparable to disks
- 3D transformations and labeled transformations
- Higher level properties: global functionality, response to stimuli, self-repair
- Distributed transformations

Open Problems

- What is the exact complexity of RotC-Transformability?
- What is the complexity of computing the optimum transformation? Can it be satisfactorily approximated?
- Consider other module-shapes, e.g., hexagons
  - Geometry changes, different actuation possibilities, not directly comparable to disks
- 3D transformations and labeled transformations
- Higher level properties: global functionality, response to stimuli, self-repair
- Distributed transformations

Open Problems

- What is the exact complexity of RotC-Transformability?
- What is the complexity of computing the optimum transformation? Can it be satisfactorily approximated?
- Consider other module-shapes, e.g., hexagons
  - Geometry changes, different actuation possibilities, not directly comparable to disks
- 3D transformations and labeled transformations
- Higher level properties: global functionality, response to stimuli, self-repair
- Distributed transformations


Thank You!