PARALLEL SELF-ASSEMBLY AND SORTING OF POLYOMINOES UNDER UNIFORM CONTROL INPUTS

A Thesis

Presented to

the Faculty of the Department of Electrical and Computer Engineering

University of Houston

in Partial Fulfillment

of the Requirements for the Degree

Master of Science

in Electrical Engineering

by
Sheryl Manzoor

December 2018

PARALLEL SELF-ASSEMBLY AND SORTING OF POLYOMINOES UNDER UNIFORM CONTROL INPUTS

Sheryl Manzoor

Approved:

Chair of the Committee
Aaron T. Becker, Assistant Professor
Department of Electrical and Computer Engineering

Committee Members:

Paul Ruchhoeft, Associate Professor
Department of Electrical and Computer Engineering

Rose Faghih, Assistant Professor
Department of Electrical and Computer Engineering

Suresh K. Khator, Associate Dean,
Cullen College of Engineering

Badrinath Roysam, Professor and Chair,
Electrical and Computer Engineering

Acknowledgements

I am highly grateful to my thesis advisor, Aaron T. Becker for his immense support in helping me accomplish my research goals. I appreciate him for always guiding me to see new research directions and for helping me in overcoming the many obstacles I faced during the process. His valuable feedback on my work ensured that I perform better and it made my research a great learning experience.

I would also like to express my gratitude to my collaborators Samuel Sheckman, Li Huang, and Jarrett Lonsford for working with me and contributing to my projects.

Last but not the least, I would like to thank my parents (Manzoor Masih and Mussarat Manzoor) and my siblings Mahwash, Salman and Faryal for their constant encouragement throughout my research.

PARALLEL SELF-ASSEMBLY AND SORTING OF POLYOMINOES UNDER UNIFORM CONTROL INPUTS

An Abstract

of a

Thesis

Presented to

the Faculty of the Department of Electrical and Computer Engineering

University of Houston

in Partial Fulfillment

of the Requirements for the Degree

Master of Science

in Electrical Engineering

by
Sheryl Manzoor

December 2018

Abstract

Automated assembly at micro and nano-scale is essential for manufacturing smaller and inexpensive products at faster rates. Traditional micro-assembly and manipulation techniques involve micro-grippers and manipulators that have limited degrees of freedom when compared to their macro-scale counterparts. A major disadvantage of these techniques is that only one part can be manufactured at a time. Thus, it would be a significant progress if control algorithms are devised that can automatically assemble a huge quantity of small scale components. We present a novel approach towards micro-assembly which employs a large swarm of micro-particles, controlled by a global signal to assemble arbitrary 2D structures. The algorithm automatically generates a micro-factory layout for a given structure, and when this layout is actuated, it manufactures the required number of copies of the shape. We have analyzed the size and time complexity of the micro-factories and present several simulation and hardware results for the assembly task. Since the assembly process is performed in open-loop, the assembled parts could have errors, and in order to detect these errors, we sort them based on their shape.

Table of Contents

List of Figures

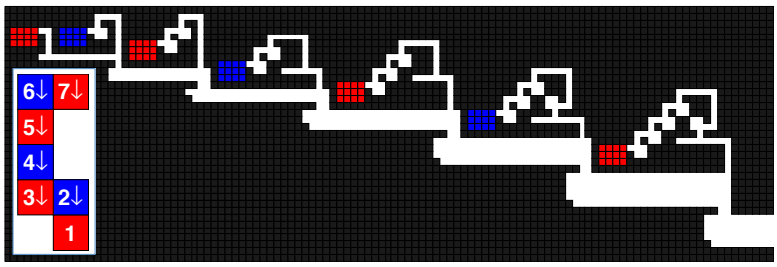
Chapter 1

Introduction

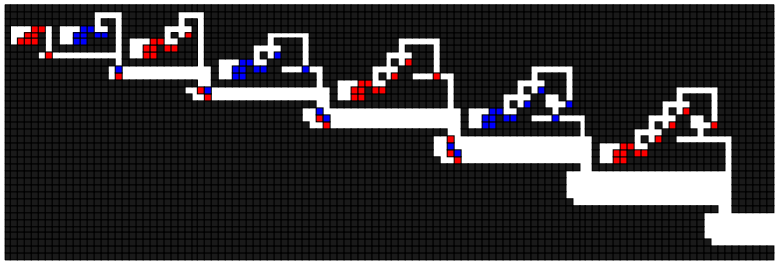
One of the exciting new directions of robotics is the design and development of micro- and nanorobot systems, with the goal of letting a massive swarm of robots perform complex operations in a complicated environment. Due to scaling issues, individual control of the involved robots becomes physically impossible: while energy storage capacity drops with the third power of robot length, medium resistance decreases much slower. As a consequence, current micro- and nanorobot systems with many robots are steered and directed by an external force that acts as a common control signal [1, 2, 3, 4, 5, 6, 7]. These common control signals include global magnetic or electric fields, chemical gradients, and turning a light source on and off.

Having only one global signal that uniformly affects all robots at once limits the swarm's ability to perform complex operations. Independent control is possible by designing heterogeneous particles that respond differently to the global input, but this approach requires precise differences in each robot and is best suited for small populations. Alternatively, control symmetry can be broken using interactions between the robot swarm and obstacles in the environment. This thesis builds on the techniques for controlling many simple particles with uniform control inputs presented in [8, 9, 10], where we demonstrated how such a system could implement digital computation. Fig. 1.1 illustrates the main contribution of this thesis: algorithms to produce a factory that uses global inputs to assemble arbitrary polyominoes. A *polyomino* is a 2D geometric figure formed by joining one or more equal squares edge to edge.

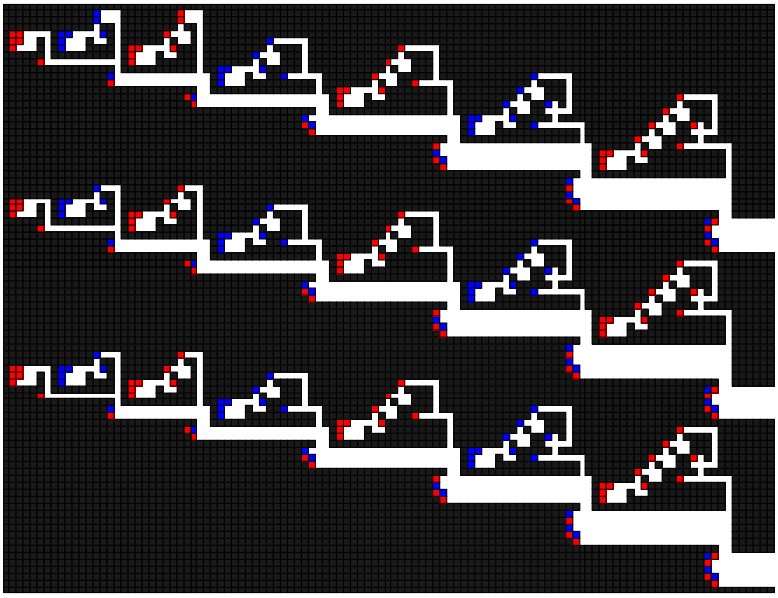
This thesis combines microscale hybrid organic/inorganic particles with novel swarm control algorithms for mask-free programmable patterning and micro-assembly. Specifi-



(a) Seven-tile polyomino factory, 0 commanded moves, 0 unit steps.



(b) Same factory, 18 commanded moves, 136 unit steps.



(c) Parallel assembly with three factories, 28 commanded moves, 221 unit steps, three complete polyominoes.

Figure 1.1: Factory schematics for assembling the seven-tile polyomino in (a). Numbers and arrows on the polyomino show the build order and direction for build. All tiles are actuated simultaneously by the same global field.

cally, this thesis applies swarm control and particle logic computations to magnetically actuate artificial cells, to use them as micro-scale robotic swarms that create complex, high resolution, 2D patterns and assemblies.

1.1 Microscale Biomanufacturing

Naturally derived biomaterials as building blocks for functional materials and devices are increasingly desired because they are often environmentally and biologically safer than purely synthetic materials. One such class of materials, polysaccharide based hydrogels, are intriguing because they can reversibly encapsulate a variety of smaller components. Many groups have termed these loaded-alginate particles *artificial cells*, because they mimic the basic structure of living cells (membrane, cytoplasm, organelles, etc.) [11, 12, 13]. Construction with these micron-sized gels has numerous applications in industry, including cell manipulation, tissue engineering, and micro-particle assembly [14, 15, 16, 17, 18], but requires fundamental research in biology, medicine, and colloidal science. While there are several methods to efficiently fabricate these particulate systems, it is still challenging to construct larger composite materials out of these units [19]. Traditional methods of assembling larger macro-scale systems are unemployable due to the change of dominant forces at small length scales. In particular, forces due to electromagnetic interactions dominate gravitational forces at the micro-scale resulting in strong adhesion and sudden shifts in the position of microparts under atmospheric conditions. To form constructs out of microgels, groups have traditionally turned to non-robotic microfluidic systems that utilize a variety of actuation methods, including mechanical, optical, dielectrophoretic, acoustophoretic, and thermophoretic [20, 21, 22, 23, 24]. While each of these methods has proven to be capable of manipulating biological cells, each method has significant drawbacks that limit their widespread application. For example, microscale mechanical, acoustophoretic, and thermophoretic manipulation methods use stimuli that can be potentially lethal to live cells [25]. Furthermore, most, if not all,

of these techniques require expensive equipment and lack control schemes necessary to precisely manipulate large numbers of cells autonomously.

1.2 Control Swarms Using Only Global Signals

Micro- and nanorobotic systems are an exciting frontier in robotics, with potential impacts in the fields of manufacturing and medicine. Chemists, biologists, and roboticists have shown the ability to produce very large populations (10^3–10^{14}) of small scale (10^{-9}–10^{-6} m) robots using a diverse array of materials and techniques [26, 27, 28]. Untethered swarms of these tiny robots may be ideal for on-site construction of high-resolution macroscale materials and devices. While these new types of large-population, small-sized, robotic systems have many advantages over their larger-scale counterparts, they also present a set of unique challenges in terms of their control. Due to current limitations in fabrication, micro- and nanorobots have little-to-no onboard computation, along with limited computation and communication ability [28, 29, 30]. These limitations make controlling swarms of these robots individually impractical. Thus, these robotic systems are often controlled by a uniform global external signal (e.g. chemical gradients, electric and magnetic fields), which makes motion planning for large robotic populations in tortuous environments difficult. At the macro-scale, automated control of devices floating in water in [31] and fluidic self assembly in [32] were presented, but as *stochastic* processes that can be controlled by turning a global signal on and off. We recently demonstrated that obstacles present in the workspace can *deterministically* break the symmetry of approximately identical robotic swarms, enabling positional configuration of robots [33]. Given sufficient free space, a single obstacle is sufficient for positional control over N particles. This method can be used to form complex assemblies out of large swarms of mobile microrobotic building blocks, using only a single global input signal.

1.3 Microrobot Based Microassembly

The ability to create microrobots, and control algorithms capable of autonomous manipulation and assembly of small scale components into functional materials is currently a major manufacturing challenge [11]. While several microrobots capable of performing simple manipulation and assembly tasks have been reported [12, 13, 14, 15, 16, 17], few have shown the ability to pattern intricate designs or assemble complex multi-component parts. Recently, groups have begun to develop cell-safe magnetically-actuated microrobotic systems for cell patterning, yet their method is limited in that these systems are manually controlled, not automated, and suffer from low spatial resolution [34, 35]. For recent advances in automated micro-assembly, see [36], but these techniques focus on a set of micro manipulators assembling one component at a time. This thesis focuses on parallelizable techniques.

1.4 Assembly Planning

Algorithm techniques for optimizing assembly operations have a rich history, see review article [37]. Our thesis determines if a polyomino has a feasible assembly sequence, similar to the planning in [38].

This thesis document has been arranged as follows: In the first chapter we discuss sequential assembly where one particle is added at a time to build a complete polyomino shape. The second chapter introduces a more efficient way of assembling 2D shapes and presents the concept of staged assembly where sub-assemblies are combined together to make a polyomino. Finally, the third chapter discusses sorting of the polyominoes based on their physical shape. Sorting is performed to detect any errors in the shape of the polyomino after the assembly process is completed. We end the thesis with some conclusions and future research directions. The work presented in this thesis is cited by [39], where they have discussed the advances possible in medicine using micro-robots and

in [40], which presents algorithms for the assembly of polyominoes in 3D.

Chapter 2

Parallel Self-Assembly of Polyominoes

This chapter is based on the journal article [41], **'Parallel Self-Assembly of Polyominoes under Uniform Control Inputs'** by *Sheryl Manzoor, Samuel Sheckman, Jarrett Lonsford, Hoyeon Kim, Min Jun Kim, and Aaron T. Becker.* My major contribution in this work is the algorithms for assembling polyominoes.

2.1 Polyomino Assembly by Global Control

This section explains how to design factories that build arbitrary-shaped 2D polyominoes. We first assign species to individual tiles of the polyomino, second discover a build path, and finally build an assembly line of factory components that each add one tile to a partially assembled polyomino and pass the polyomino to the next component.

2.1.1 Model

The model for assembly has following rules:

(1). A planar grid *workspace* W is filled with a number of unit-square particles (each occupying one cell of the grid) and some fixed unit-square blocks. Each unit square in the workspace is either *free*, which a particle may occupy or *obstacle* which a particle may not occupy. Each square in the grid can be referenced by its Cartesian coordinates $x = (x, y)$.

(2). All particles are commanded in unison: the valid commands are "Go Up" (u), "Go Right" (r), "Go Down" (d), or "Go Left" (l).

(3). Particles all move until they hit an obstacle, hit a stationary particle, or share an

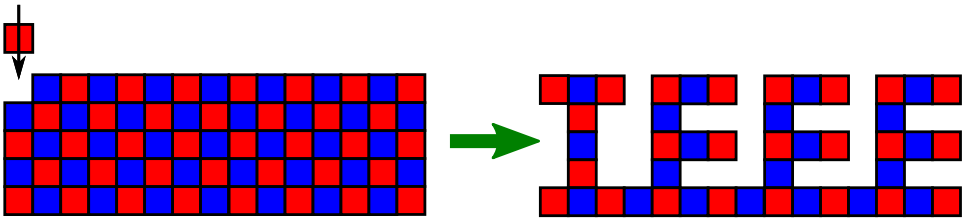


Figure 2.1: Any polyomino can be constructed with two compatible robot species that only stick to each other, shown here with red and blue tiles.

edge with a compatible particle. If a particle shares an edge with a compatible particle the two particles bond and from then on move as a unit.

(4). In this chapter, we use *cycles* of movement commands in the order $\langle r, d, l, u \rangle$. We assume the area of W is finite and issue each command long enough for the particles to reach their maximum extent.

2.1.2 Arbitrary 2D Shapes Require Two Particle Species

Polyominoes have *four-point connectivity*: a 4-connected square is a neighbor to every square that shares an edge with it.

Lemma 1. *Any polyomino can be constructed using just two species*

Proof. Label a grid with an alternating pattern like a checkerboard. Any desired polyomino can be constructed on this checkerboard, and all joints are between dissimilar species. An example shape is shown in Fig. 2.1. Red and blue colors are used to indicate particles of different species. □

The sufficiency of two species to construct any shape gives many options for implementation. The two species could correspond to any gendered connection, including ionic charge, magnetic polarity, or hook-and-loop type fasteners. Large populations of these two species can then be stored in separate hoppers and, like two-part epoxy, only assemble when dissimilar particles come in contact.

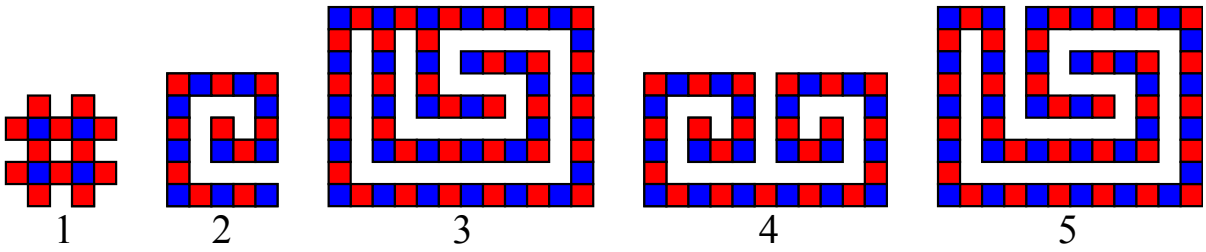


Figure 2.2: Polyomino parts. Assembly difficulty increases from left to right.

2.1.3 Complexity Handled in This Thesis

2D part geometries vary in difficulty. Fig. 2.2 shows parts with increasing complexity.

Label the first particle in the assembly process the *seed particle*. Part 1 is shaped as a '#' symbol. Though it has an interior hole, any of the 16 particles could serve as the seed particle, and the shape could be constructed around it. The second shape is a spiral, and must be constructed from the inside-out. If the outer spiral was completed first, there would be no path to add particles to finish the interior because added particles would have to slide past compatible particles. Increasing the number of species would not solve this problem, because there is a narrow passage through the spiral that forces incoming parts to slide past the edges of all the bonded particles. The third shape contains a loop, and the interior must be finished before the loop is closed. Shape 4 is the combination of a left-handed and a right-handed spiral. Adding one particle at a time in 2D cannot assemble this part, because each spiral must be constructed from the inside-out. Instead, this part must be divided into sub-assemblies that are each constructed, and then combined. Shape 5 contains compound overhangs, and its assembly requires a workspace with a variety of obstacles, recently discovered by [42]. The algorithms in this letter detect if the desired shape can be constructed one particle at a time. If so, a build order is provided, and a factory layout is designed.

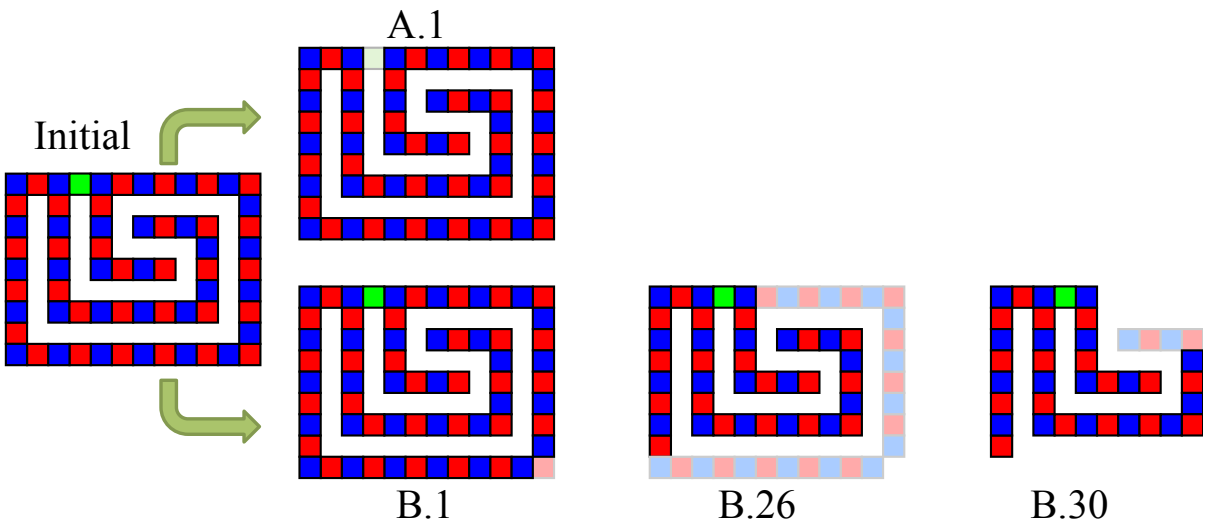


Figure 2.3: In the top row the green tile is removed first, resulting in a polyomino that cannot be decomposed. However, if the bottom right tile is removed first, deconstruction is possible.

2.1.4 Discovering a Build Path

Given a polyomino, Alg. 1 determines if the polyomino can be built by adding one component at a time. The problem of determining a build order is difficult because there are $O(n!)$ possible build orders, and many of them may violate the constraints given in Section 2.1.1. Each new tile must have a straight-line path to its goal position in the polyomino that does not collide with any other tile, does not slide past an opposite specie tile, and terminates in a mating configuration with an opposite specie tile. However, as in many robotics problems, the inverse problem of deconstruction is easier than the forward problem of construction.

Algorithm 1 FindBuildPath($\mathbf{P}$)

$\mathbf{P}$ is the x, y coordinates of a 4-connected polyomino. Returns $\mathbf{C}$, $\mathbf{c}$ and $\mathbf{m}$ where $\mathbf{C}$ contains sequence of polyomino coordinates, $\mathbf{c}$ is a vector of color labels, and $\mathbf{m}$ is a vector of directions for assembly.

1: $\mathbf{c} \leftarrow$LabelColor($\mathbf{P}$)
2: $\{\mathbf{C}, \mathbf{m}\} =$Decompose($\mathbf{P}, \mathbf{c}$)
3: **return** $\{\mathbf{C}, \mathbf{c}, \mathbf{m}\}$

Alg. 1 first assigns each tile in the polyomino a color, then calls the recursive function DECOMPOSE, which returns either a build order of polyomino coordinates and the directions to build, or an empty list if the part cannot be constructed. DECOMPOSE starts by calling the function ERODE. ERODE first counts the number of components in the 8-connected freespace. An 8-connected square is a neighbor to every square that shares an edge or vertex with it. If there is more than one connected component, the polyomino contains loops. ERODE maintains an array of the remaining tiles in the polyomino $\mathbf{R}$. In the inner *for loop* at line 8, a temporary array $\mathbf{T}$ is generated that contains all but the jth tile in $\mathbf{R}$ sorted by the number of neighbors so a tile with one neighbor is checked before tiles with two or three. This *for loop* simply checks (1) if the jth tile can be removed along a straight-line path without colliding with any other particle or sliding past an opposite specie tile in line 9, (2) that its removal does not fragment the remaining polyomino into more than one piece in line 10, and (3) that its removal does not break a loop in line 11. If no loops are present, this algorithm requires at most $n/2(1+n)$ iterations, because there are n particles to remove, and each iteration considers one less particle than the previous iteration.

Polyominoes with loops require care, because decomposing them in the wrong order can make disassembly impossible, as shown in Fig. 2.3. Deconstruction order matters if loops are present. Loops occur when the 8-connected freespace has more than one connected component. If loops exist then ERODE may return only a partial decomposition, so DECOMPOSE must then try every possible break point and recursively call DECOMPOSE until either a solution is found, or all possible decomposition orders have been tested. The worst-case number of function calls of DECOMPOSE are proportional to the factorial of the number of loops, $O(|8\text{-}\textsc{ConnComp}(\neg\mathbf{P})|!)$. Though large, this is much less than $O(n!)$.

Algorithm 2 ERODE($\mathbf{P}, \mathbf{c}$)

$\mathbf{P}$ is the x, y coordinates of a 4-connected polyomino and $\mathbf{c}$ is a vector of color labels. Returns $\mathbf{R}$, $\mathbf{C}$, $\mathbf{m}$, and ℓ where $\mathbf{R}$ is a list of coordinates of the remaining polyomino, $\mathbf{C}$ contains sequence of tile coordinates that were removed, $\mathbf{m}$ is a vector of directions for assembly, and ℓ if loops were encountered. $\mathbf{d} \leftarrow \{r, d, l, u\}$

$\mathbf{C} \leftarrow \{\}, \mathbf{m} \leftarrow \{\}, \ell \leftarrow$ FALSE, $\mathbf{R} \leftarrow \mathbf{P}$
$w \leftarrow |8\text{-CONNCOMP}(\neg\mathbf{R})|$
while $1 < |\mathbf{R}|$ **do**
    successRemove $\leftarrow$ FALSE
    $\mathbf{R} \leftarrow$SORT($\mathbf{R}$) ▷ sort by number of neighbors
    for $j \leftarrow 1, j \leq |\mathbf{R}|$ **do**
        $\mathbf{p} \leftarrow \mathbf{R}_j, \mathbf{T} \leftarrow \mathbf{R} \backslash \mathbf{R}_j$
        for $k \leftarrow 1, k \leq 4$ **do**
            if CHECKPATHTILE($\mathbf{T}, \mathbf{p}, \mathbf{d}_k, \mathbf{c}$) **and**
                $1 = |4\text{-CONNCOMP}(\mathbf{T})|$ **then**
                if $w = |8\text{-CONNCOMP}(\neg\mathbf{T})|$ **then**
                    $\mathbf{R} \leftarrow \mathbf{T}$, *successRemove* $\leftarrow$ TRUE
                    $\mathbf{C}_{1+|\mathbf{R}|} \leftarrow \mathbf{p}, \mathbf{m}_{|\mathbf{R}|} \leftarrow \mathbf{d}_k$
                else $\ell \leftarrow$ TRUE
                break
    if *successRemove* = FALSE **then**
        $\mathbf{C} \leftarrow \{\}, \mathbf{m} \leftarrow \{\}$
        break
if $|\mathbf{R}| = 1$ **then**
    $\mathbf{C}_1 \leftarrow \mathbf{R}_1$
return $\{\mathbf{R}, \mathbf{C}, \mathbf{m}, \ell\}$

2.1.5 Hopper Construction

Two-part adhesives react when components mix. Placing components in separate containers prevents mixing. Similarly, storing many particles of a single specie in separate containers allows controlled mixing.

We can design *part hoppers*, containers that store similarly labelled particles. These particles will not bond with each other. The hopper shown in Fig. 2.4 releases one particle every cycle. Delay blocks are used to ensure the nth part hopper does not start releasing particles until cycle n. For ease of exposition, this chapter uses a unique hopper for each tile position. This enables precise positioning of different materials, but a particle logic

Algorithm 3 DECOMPOSE($\mathbf{P}, \mathbf{c}$)

$\mathbf{P}$ is the x, y coordinates of a 4-connected polyomino and $\mathbf{c}$ is a vector of color labels. Returns $\mathbf{C}$ and $\mathbf{m}$ where $\mathbf{C}$ contains sequence of polyomino coordinates and $\mathbf{m}$ is a vector of directions for assembly. $\mathbf{d} \leftarrow \{u, d, l, r\}$

$\{\mathbf{R}, \mathbf{C}, \mathbf{m}, \ell\} \leftarrow$ERODE($\mathbf{P}, \mathbf{c}$)
if $|\mathbf{R}| = 0$ **or** $\neg\ell$ **then**
    return $\{\mathbf{C}, \mathbf{m}\}$
for $j \leftarrow 1, j \leq |\mathbf{R}|$ **do**
    $\mathbf{p} \leftarrow \mathbf{R}_j, \mathbf{T} \leftarrow \mathbf{R} \backslash \mathbf{R}_j$
    for $k \leftarrow 1, k \leq 4$ **do**
        if (CHECKPATHTILE($\mathbf{T}, \mathbf{p}, \mathbf{d}_k, \mathbf{c}$) **and**
            $1 = |4\text{-CONNCOMP}(\mathbf{T})|$) **then**
            $\{\mathbf{C2}, \mathbf{m2}\} \leftarrow$DECOMPOSE($\mathbf{T}, \mathbf{c}$)
            if $\mathbf{C2} \neq \{\}$ **then**
                $\mathbf{C}_{1:|\mathbf{C2}|+1} \leftarrow \{\mathbf{C2}, \mathbf{p}\}$
                $\mathbf{m}_{1:|\mathbf{m2}|+1} \leftarrow \{\mathbf{m2}, \mathbf{d}_k\}$
                return $\{\mathbf{C}, \mathbf{m}\}$
            break
return $\{\mathbf{C} \leftarrow \{\}, \mathbf{m} \leftarrow \{\}\}$

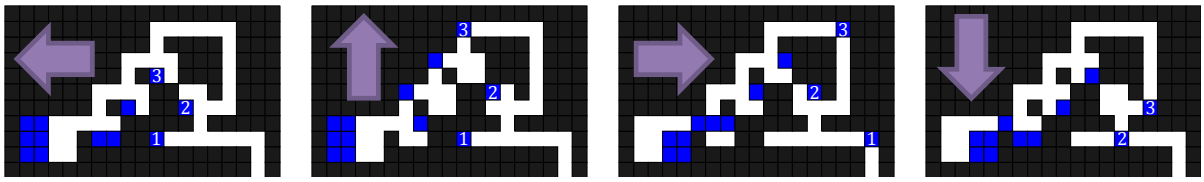


Figure 2.4: Hopper with five delays. The hopper is filled with similarly-labelled robots that will not combine. Every clockwise command cycle releases one robot from the hopper.

system could use just two hoppers, similar to our particle logic systems in [9].

2.1.6 Part Assembly Jigs

Assembly is an iterative procedure. A factory layout is generated by BUILDFACTORY($\mathbf{P}, n_c$), described in Alg. 4. This function takes a 2D polyomino $\mathbf{P}$ and, if $\mathbf{P}$ has a valid build path, designs an obstacle layout to generate n_c copies of the polyomino. A polyomino is composed of $|\mathbf{P}| = n$ tiles.

For each tile, the function FACTORYADDTILE ($n_c, \mathbf{b}, m, C, c, w$) described in Alg. 5

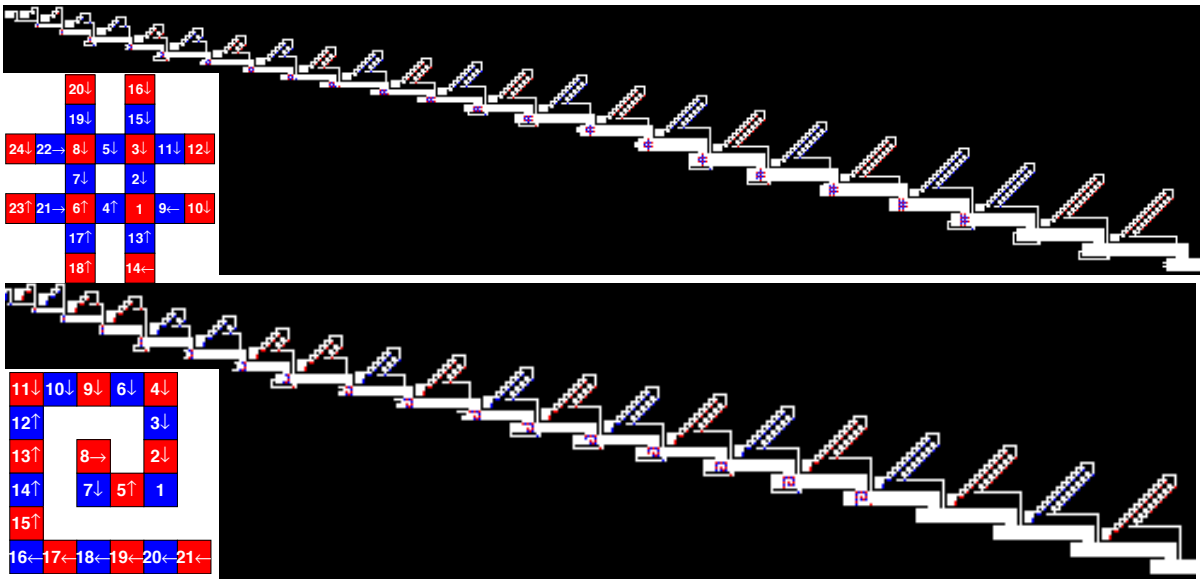


Figure 2.5: (Top) A twenty-four tile factory, step 82 for a '#' shape. (Bottom) A twenty-one tile factory, step 66 for a spiral (zoom in for details in this vector graphic).

is called to generate an obstacle configuration $\mathbf{A}$. $\mathbf{A}$ forms a hopper that releases a particle each iteration and a chamber that temporarily holds the partially-assembled polyomino $\mathbf{b}$ and guides the new particle C to the correct mating position. A 24-tile factory is shown in Fig. 2.5.

Algorithm 4 BUILDFACTORY$(\mathbf{P}, n_c)$

$\mathbf{P}$ is the x, y coordinates of a 4-connected polyomino. n_c is the number of parts desired. Returns a two dimensional array $\mathbf{F}$ containing the factory obstacles and filled hoppers.

```
F ← {}                                              ▷ the factory obstacle array
{C, c, m} ← FINDBUILDPATH(P)
if {} = m then
    return F
{A, b} ←FACTORYFIRSTTILE(n_c, c_i, w)
for i ← 2, i ≤ |c|) do
    {A, b} ←FACTORYADDTILE(n_c, b, m_{i−1}, C_i, c_i, w)
    F ←CONCATFACTORIES(F, A)
return F
```

Algorithm 5 FACTORYADDTILE($n_c, \mathbf{b}, m, C, c, w$)

1: $\{\mathbf{hopper}\} \leftarrow$HOPPER$(c, n_c, w)$
2: **if** $m = d$ **and** $(C_x \leq \max \mathbf{b}_x$ **or** $C_y < \min \mathbf{b}_y)$ **then**
3: $\quad \{\mathbf{A}, \mathbf{b}\} \leftarrow$DOWNDIR$(\mathbf{hopper}, \mathbf{b}, \mathbf{C})$
4: **else if** $m = l$ **and** $(C_y \leq \max \mathbf{b}_y$ **or** $C_x > \max \mathbf{b}_x)$ **then**
5: $\quad \{\mathbf{A}, \mathbf{b}\} \leftarrow$LEFTDIR$(\mathbf{hopper}, \mathbf{b}, \mathbf{C})$
6: **else if** $m = l$ **and** $(C_x \geq \max \mathbf{b}_x$ **or** $C_y > \max \mathbf{b}_y)$ **then**
7: $\quad \{\mathbf{A}, \mathbf{b}\} \leftarrow$UPDIR$(\mathbf{hopper}, \mathbf{b}, \mathbf{C})$
8: **else if** $m = r$ **and** $(C_y \geq \min \mathbf{b}_y$ **or** $C_x < \min \mathbf{b}_x)$ **then**
9: $\quad \{\mathbf{A}, \mathbf{b}\} \leftarrow$RIGHTDIR$(\mathbf{hopper}, \mathbf{b}, \mathbf{C})$
10: **return** $\{\mathbf{A}, \mathbf{b}\}$

2.2 Analysis

This section analyzes the travel distance and space required for a factory and gives simulation results. Algorithms 1—5 were coded in MATLAB and are available at [43].

2.2.1 Maximum Distance Travelled

Running a factory simulation has three phases: (1) ramp up, (2) production, and (3) wind down. During the $n-1$ *ramp up* cycles, the first polyomino is being constructed one tile at a time and no polyominoes are produced. Clever design of delays in the part hoppers ensures no unconnected tiles are released. During *production* cycles, one polyomino is finished each cycle. Once the first part hopper empties, the $n-1$ *wind down* cycles each produce a complete polyomino as each successive hopper empties. This section analyzes maximum distance, defined as the maximum distance any tile must move. There are two results, *construction distance*, the maximum distance required to assemble a single polyomino from scratch, and *cycle distance*, the maximum distance required during production cycles to advance all partial assemblies one cycle. Since a polyomino contains n tiles, the *construction distance* during production cycles is $n \cdot$(*cycle distance*).

Cycle distance is the sum of the maximum distances moved in each direction. As

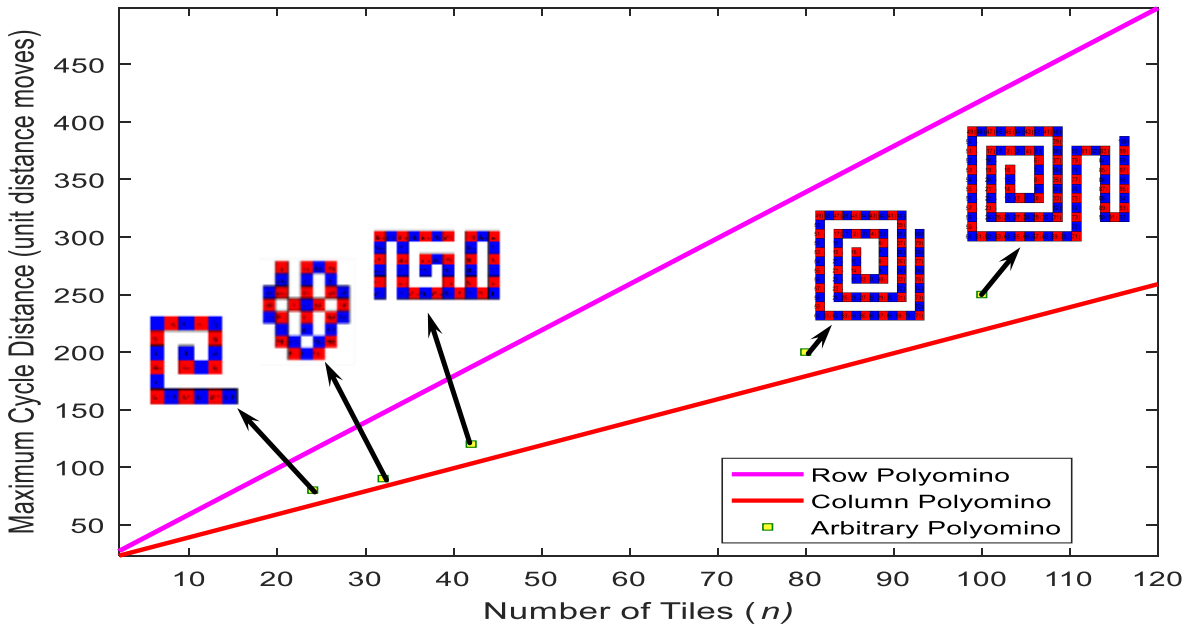


Figure 2.6: Worst-case cycle distance plotted as a function of polyomino size n.

shown in Fig. 2.6, polyominoes shaped as a $n \times 1$ row require the longest distance of $4n + 16$. Polyominoes shaped as a $1 \times n$ column require the least distance of $2n + 16$. Construction distance therefore requires $O(n^2)$ distance.

2.2.2 Space Required

The space required by a factory is a function of the widths of individual sub-factories and height of the last sub-factory. The first sub-factory is constructed separately and it does not have any delay. Beginning from the second sub-factory, height can be computed as a function of the number of copies n_c of the polyomino, width of the hopper w, position of the sub-factory i, and rows of the sub-assembled polyomino $\mathbf{b}_y$ as in (2.1):

$$height(i) = \left\lceil \frac{n_c}{w} \right\rceil + 2\left(\left\lceil \frac{i}{2} \right\rceil + \mathbf{b}_y\right) + \begin{cases} 4, & \text{for } m = l \text{ or } d, i \geq 2 \\ 7, & \text{for } m = u \text{ or } r, i \geq 2. \end{cases} \tag{2.1}$$

If a tile is added before the top row of $\mathbf{b}$, then an additional row is added to the height. The width of the sub-factory can be calculated using (2.2):

$$width(i) = width_{\text{hopper+delays}} + \begin{cases} (\mathbf{b}_x - column_{\text{loc}}), & \text{for } m = d \\ 0 & \text{for } m \neq d, \end{cases} \tag{2.2}$$

where

$$width_{\text{hopper+delays}} = w + 2\left\lceil \frac{i}{2} \right\rceil + 8, i \geq 2. \tag{2.3}$$

Note: The symbol $\lceil a \rceil$ is the ceiling of a.

In a case where twice of $\mathbf{b}_x$ is greater than $width_{\text{hopper+delays}}$, then additional columns are added to the left of the sub-factory. When a tile is added to $\mathbf{b}$ using a down move, width also depends on the location of the column, $column_{\text{loc}}$, to which the tile is added.

Because a factory requires $O(n)$ rows and $O(n)$ columns, the total required space is $O(n^2)$. As shown in Fig. 2.7, the required size is upper bounded by column-shaped polyominoes and lower bounded by row-shaped polyominoes, and is $O(n^2)$.

2.3 Experiment

To demonstrate Algs. 1–5, we developed two platforms at two size scales. The macro-scale demonstration board uses gravity as the external force and magnetic attraction between red and blue particles for assembly, and the micro-scale setup uses a magnetic control stage with alginate micro-particles.

2.3.1 Macro-scale, Gravity-Based Prototype

The gravity-based model shown in Fig. 2.8 uses a white workspace, red sliders for particles with magnetic north out, blue sliders for particles with magnetic south out, and

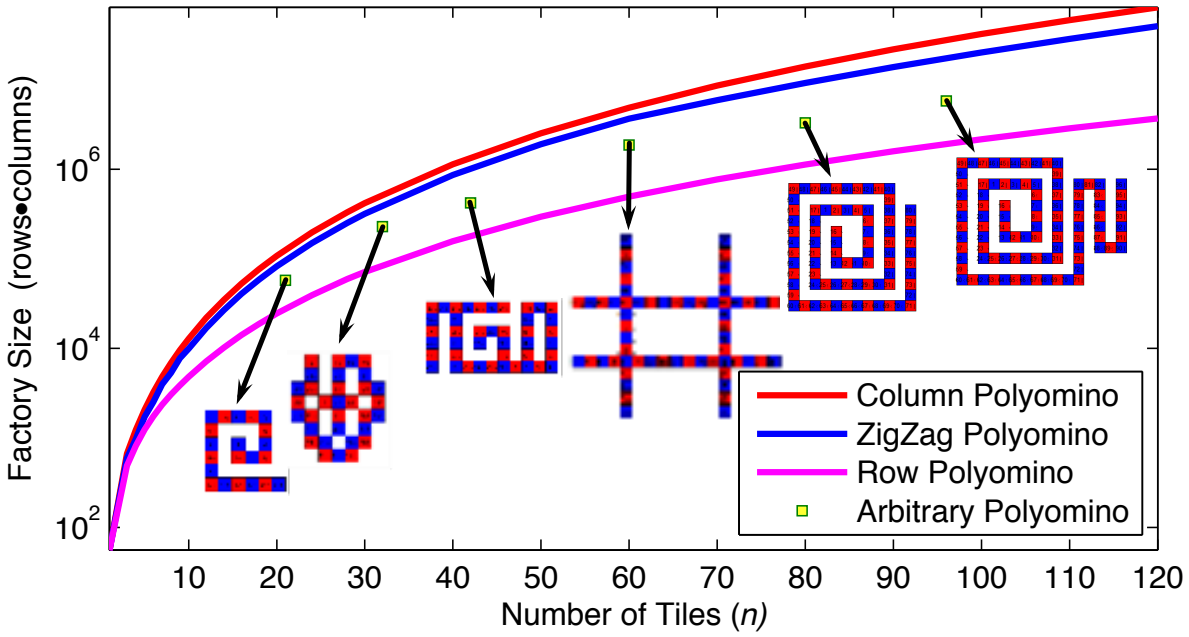


Figure 2.7: Factory size grows quadratically with the number of tiles.

black stop blocks for workspace obstacles. This model uses gravity as a global input to manipulate the red and blue sliders.

Construction and assembly: The macro-scale, reconfigurable, gravity-based model used to demonstrate parallel assembly was manufactured from laser cut acrylic, plastic dowel rods, and 3.2×3.2×1.6 mm^3 neodymium magnets. The workspace was made from a 0.6 by 0.3 meter sheet of 6.35 mm thick white acrylic. A laser cutter was used to make a grid of slider tracks 3.25 mm deep and 3.25 mm wide in the workspace as well as four holes with a diameter of 3.2 mm around each intersection of the grid for stop blocks to be securely placed. The stop blocks are made of similar black acrylic with four plastic dowel rods so they may be securely placed onto the workspace. The particles were made from similar red and blue acrylic sheets and are approximately 25 mm in diameter. The sliders have eight laser cut slots to house the magnets and have a small plastic dowel rod inserted in the center to ensure the sliders follow the tracks of the workspace.

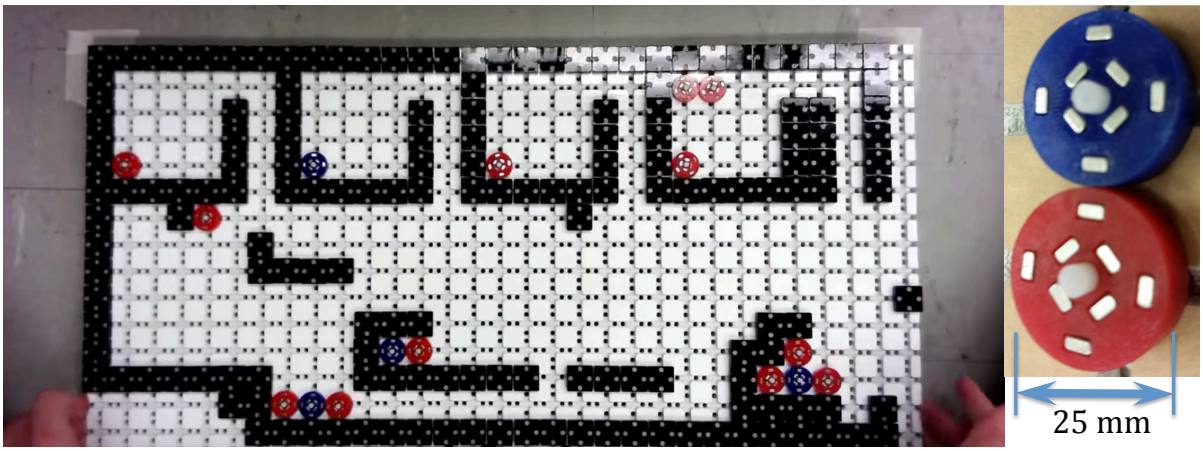


Figure 2.8: A macro-scale demonstration of particle assembly using gravity as the external force and magnetic attraction between red and blue particles for assembly.

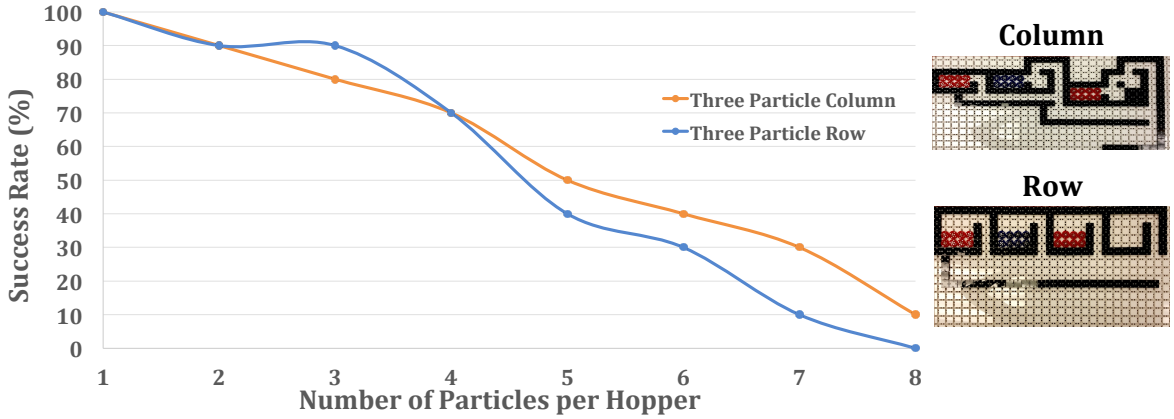


Figure 2.9: Results from assembly of macro-scale, three tile row and column polyominoes. Each data point represents 10 trials.

Forces involved: When the macro-scale demonstration is tilted at an angle of 20° most of the sliders will break free from the average static friction force of 0.0074 N and move across the workspace. At this angle the average force of weight contributing to the motion of the sliders is 0.0092 N, just enough to overcome the friction. Since the average magnetic breaking strength of the sliders is 0.1 N, sliders of opposite charge should be able to connect and overcome the force of motion of the sliders. However, there are instances where this connection does not overcome the force of motion due to a high tilt angle needed to break static friction.

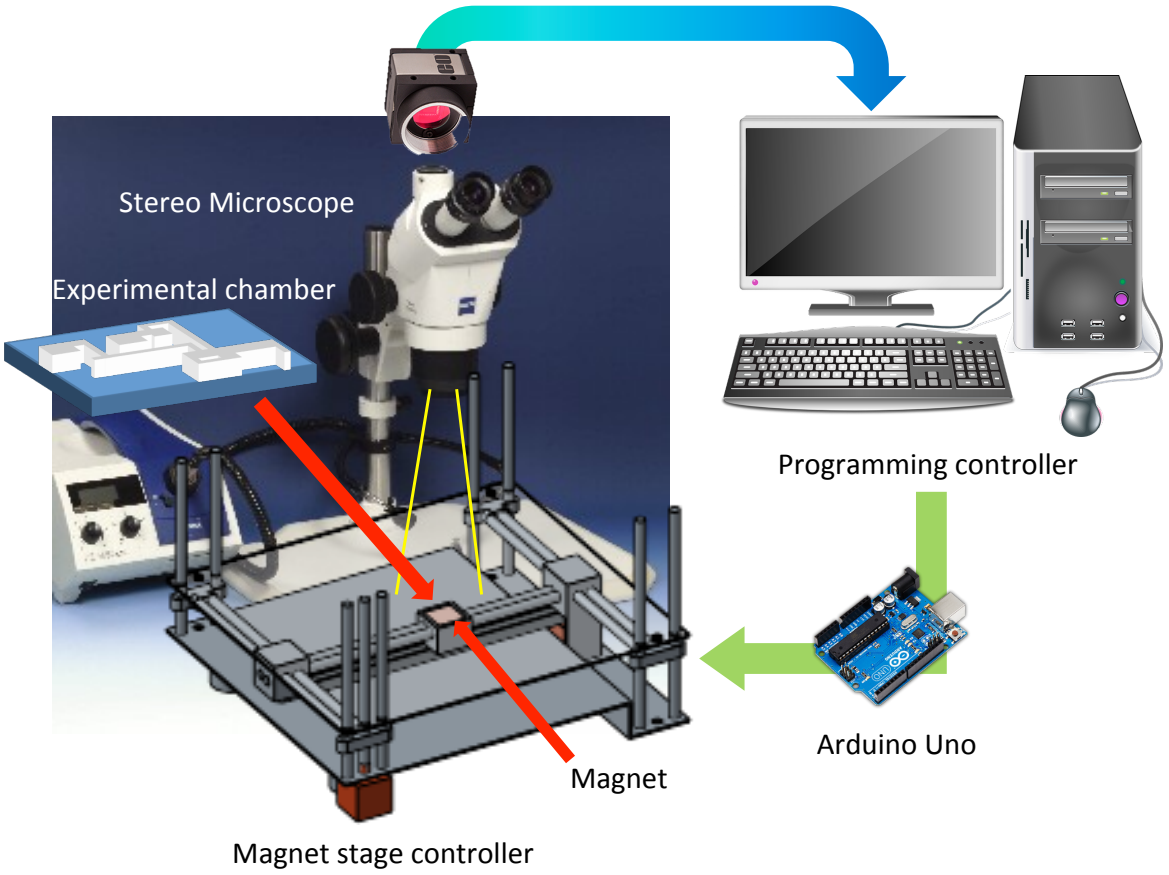


Figure 2.10: Experimental platform.

Macro Scale Results: Fig. 2.9 shows results of experimentation for three tile row and three tile column polyominoes. Success rate is high when the number of sliders in each hopper is small. This is because the system was designed for a small number of particles and the magnetic repulsion of like particles can misalign the sliders.

2.3.2 Micro-scale, Magnetic-Based Prototype

We designed a custom magnetic control stage to generate the global control inputs. This stage generates a magnetic drag force by moving a permanent magnet.

Experimental setup: Figure 2.10 shows a system schematic. The permanent magnet can translate in x and y-axes, actuated by stepper motors and moving on linear rails. The neodymium permanent magnet field strength is 1.32 T and dimensions are 50.8 $\times$ 50.8 mm^2 (K&J Magnetics). The microfluidic factory layout produced for this experiment was fabricated through traditional photolithography methods. A silicon wafer was selected as the microfabrication substrate. SU-8 2150 photoresist (MicroChem) was then spin coated onto the substrate, giving a thickness of 300 μm. The channel width is 500 μm. Channels were then filled with motility buffer composed of Dionized Water and 10% Tween 20. All microrobots used for these experiments were loaded alginate paramagnetic hydrogels, otherwise known as artificial cells. Alginate microrobots can encapsulate both organic and non-organic materials, which makes them the best suited form of microrobots to create different types of species. The alginate microrobots were fabricated through centrifugal method and these particles had diameter:

$$d_p = \sqrt[3]{(6d_n\sigma_p)/(\rho_p g)}, \tag{2.4}$$

where d_n, σ_p, ρ_p, and g are the diameter of the nozzle, surface tension of the alginate solution, density of alginate solution, and the applied gravitational force, respectively. The surface tension of alginate is 65.46 mN/m, and a density of 1.1 g/cm^3. The average microrobot size is 300 μm, and were composed of a concentration of 5% (w/v) Alginate-Na and 5% (w/v) concentration of $CaCl_2$, and then encapsulated with 10% (w/v) nano-paramagnetic particles (Iron oxide, Sigma-Aldrich). Alginate microrobots were transported at each hopper in the microfluidic factory layout, by way of a pipette. To show the process, one alginate particle was loaded in each hopper. The experimental channel was placed at the center of the stage with the magnet centered beneath the microfluidic factory layout. This position was saved as the home position for the permanent magnet. Stepper motors controlled the stage position. An Arduino UNO programmed in C++ commanded these motors using a 2 Hz control loop. After a command was initiated, such as each direction in the $\langle r, d, l, u\rangle$ cycle, the permanent magnet returned to the

home position. A non-zero magnetic gradient in the horizontal plane is only generated when the magnet moves out of its home position. The layout was observed through a stereomicroscope and the installed camera (Motion Pro X3) captured the procedure at 30 fps. The observed field of view at 0.65× magnification is 23.6×18.9 mm^2.

Experimental result: Using a factory layout generated by Alg. 4, we demonstrated micro-scale assembly using multiple alginate microrobots. The initial scene is shown in Fig. 2.11(a). The first assembly operation was then orchestrated by moving the magnet in a clockwise direction, following the $\langle r, d, l, u \rangle$ cycle as indicated in Fig. 2.11(b-d). Each input was applied sufficiently long to ensure all alginate microrobots touched a wall. Completion of the square polyomino is shown in Fig. 2.11(e).

The micro scale experiments presented in this chapter are slow and few trials were performed to assemble the polyomino. In the next chapter, these limitations are improved as the experiments are repeated several times and it takes less time to perform each experiment.

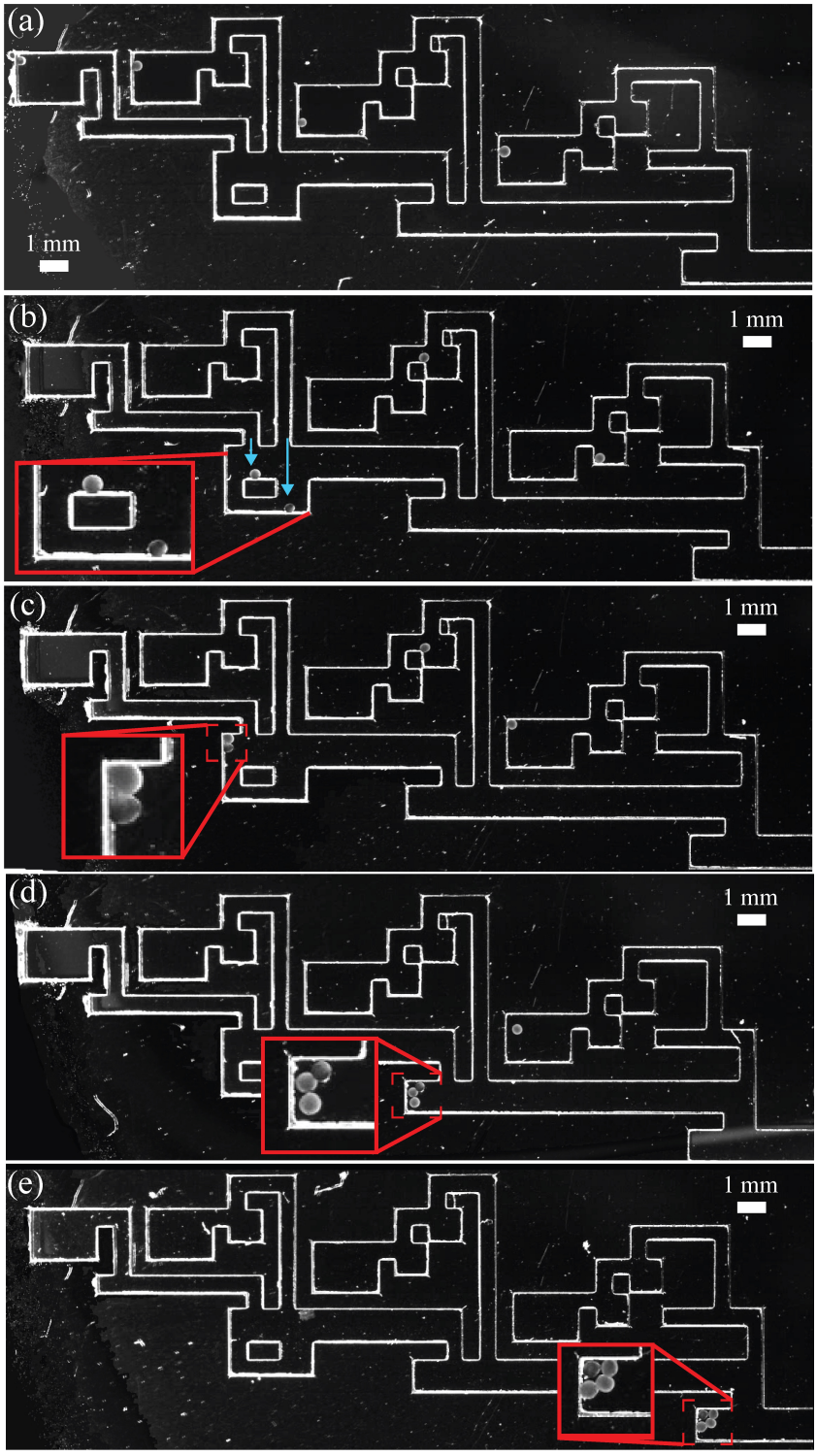


Figure 2.11: Experimental results of Alg. 4.

Chapter 3

Efficient Parallel Self-Assembly of Polyominoes

This chapter is based on the journal article [44], **'Efficient Parallel Self-Assembly of Polyominoes under Uniform Control Inputs'** by *Arne Schmidt, Sheryl Manzoor, Li Huang, Aaron T. Becker, and Sándor P. Fekete.* I contributed to hardware experimentation for this project.
Fig 3.1, Fig. 3.2, Fig. 3.3 and, Fig 3.6 are from [44].

3.1 Staged Assembly of Polyominoes

In this chapter, we look at staged assembly where sub-assemblies are combined successively to build a polyomino shape. This is a more efficent approach than assembling particles sequentially to form a polyomino because it will enable sublinear construction times. Specifically, we discuss staged assembly of convex polyominoes. A polyomino is *row convex* if the intersection of any horizontal line with the polyomino does not have more than two connected components. A *column convex* plyomino can be defined similarly. Fig 3.4 and Fig 3.5 show examples of *row convex* and *column convex* polyominoes respectively. In [44], it is shown that any convex polyomino can be assembled in six movement steps as in Fig 3.1. Furthermore, a convex polyomino can be decomposed into subpolyominoes which are all convex and then assembled to build the complete polyomino. A polyomino, can be decomposed using *straight cuts* as in 3.6. A straight cut has no turns and it divides the polyomino is a way that the resulting subpolyominoes can be pulled apart in opposite directions without blocking each other. Once the polyomino is divided, two subpolyominoes can be assembled in each cycle using the combining gadgets in Fig. 3.2 and Fig. 3.3.

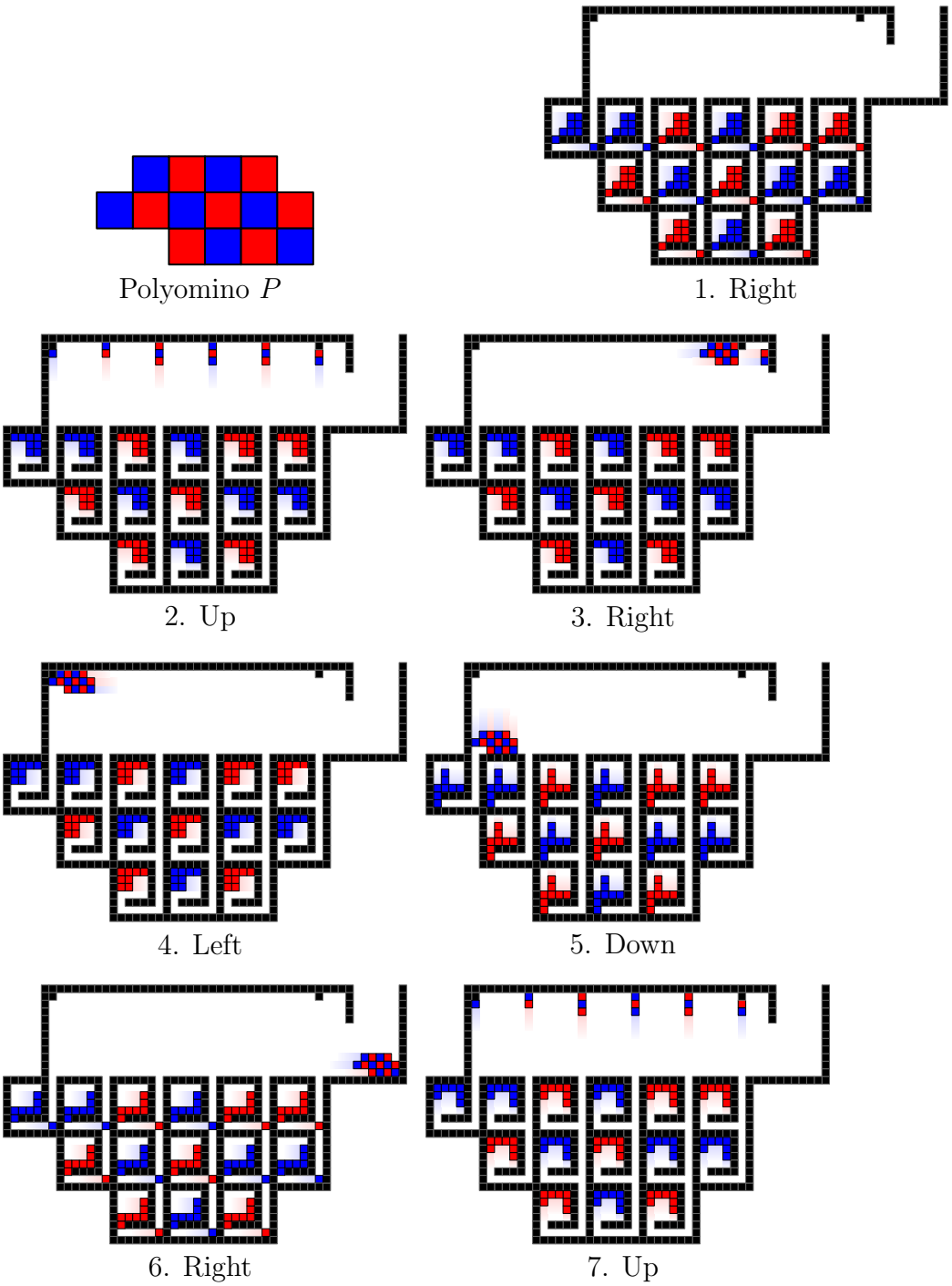
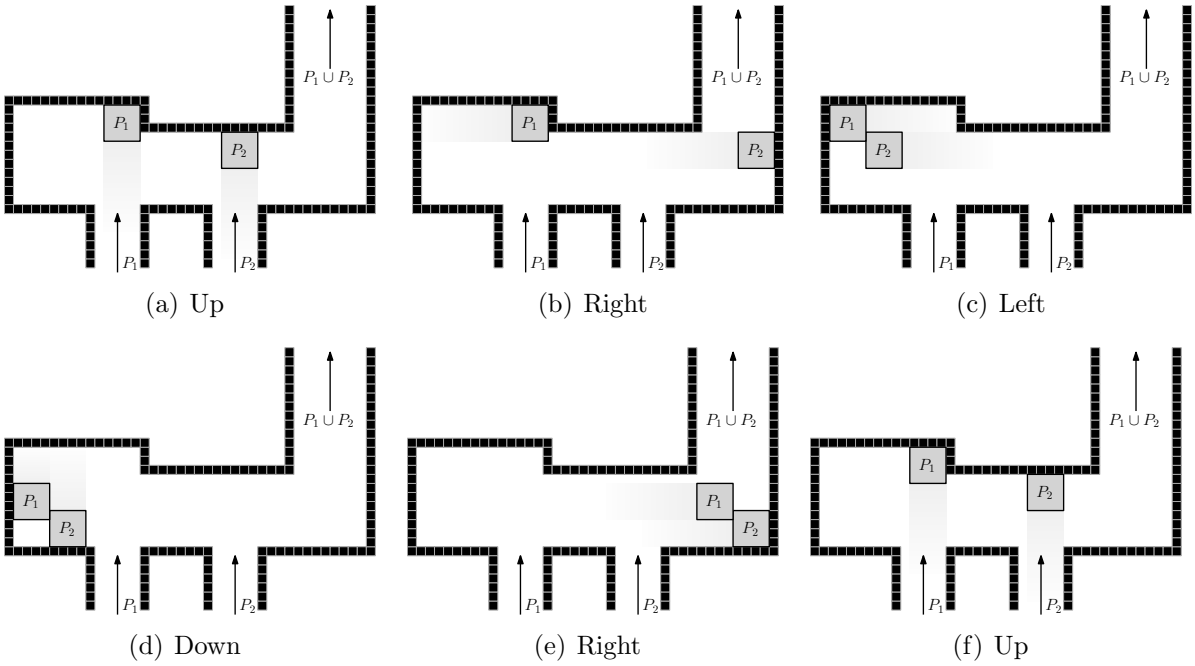


Figure 3.1: Convex polyominoes can be assembled in six movement steps. A copy of the polyomino P is released every five steps after the first copy.



(a) Up (b) Right (c) Left

(d) Down (e) Right (f) Up

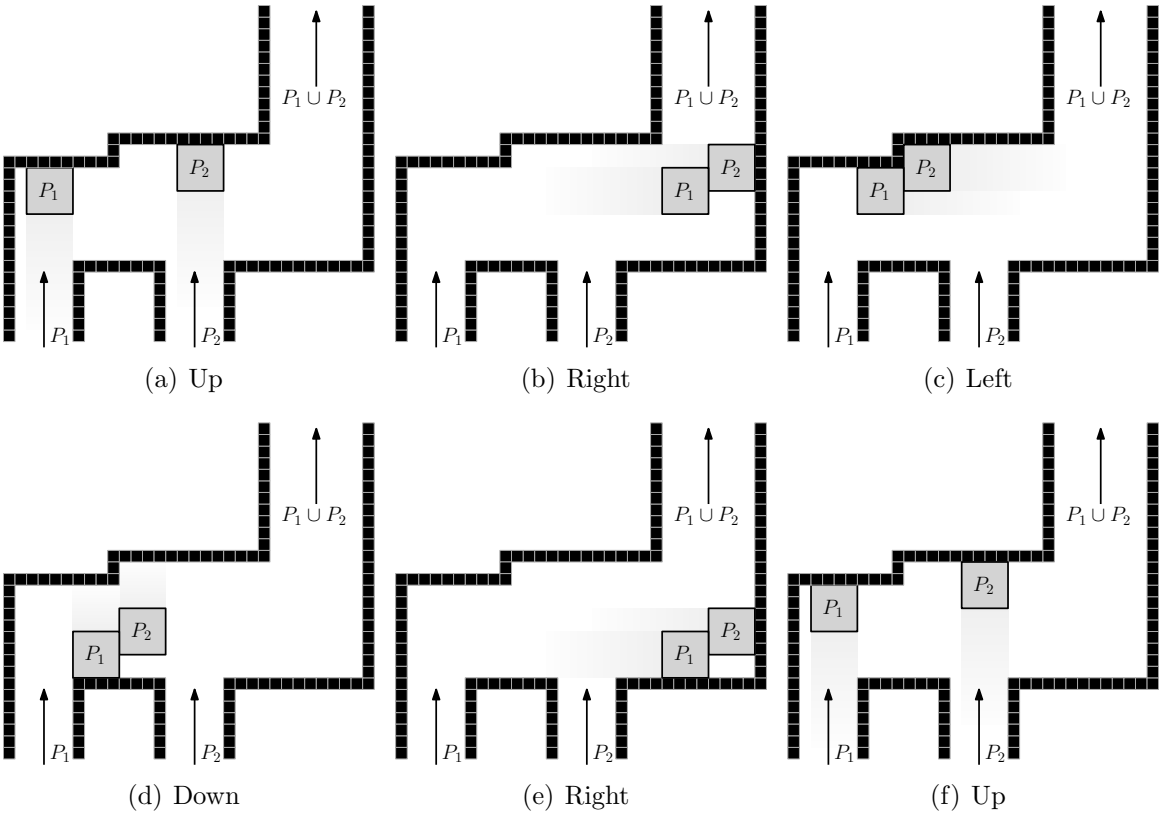
Figure 3.2: Assembling two subpolyominoes P_1 and P_2, where the topmost tile of P_1 lies above the topmost tile of P_2. These are the same movements as seen in Fig. 3.1.

3.2 Experimental Demonstration

We implemented the algorithms for staged assembly at micro and milli scale. A customized setup was used to generate a magnetic field to manipulate the magnetic particles.

3.2.1 Experimental Platform

The magnetic setup used for the experiments is shown in Fig. 4.3, consisting of two orthogonal pairs of coils with separation distance equivalent to the outer diameter (127.5 mm) of a coil. The coils (18 AWG, 1200 turns, Custom Coils, Inc) are actuated by six SyRen10-25 motor drivers, and Tekpower HY3020E is used for DC power supply. The electromagnetic platform can provide up to 101 Gauss uniform magnetic fields, and up to 150 mT/m gradient fields along any horizontal direction in the center of the



(a) Up (b) Right (c) Left

(d) Down (e) Right (f) Up

Figure 3.3: Assembling two subpolyominoes P_1 and P_2, where the topmost tile of P_2 lies above the topmost tile of P_1. These are the same movements as seen in Fig. 3.1.

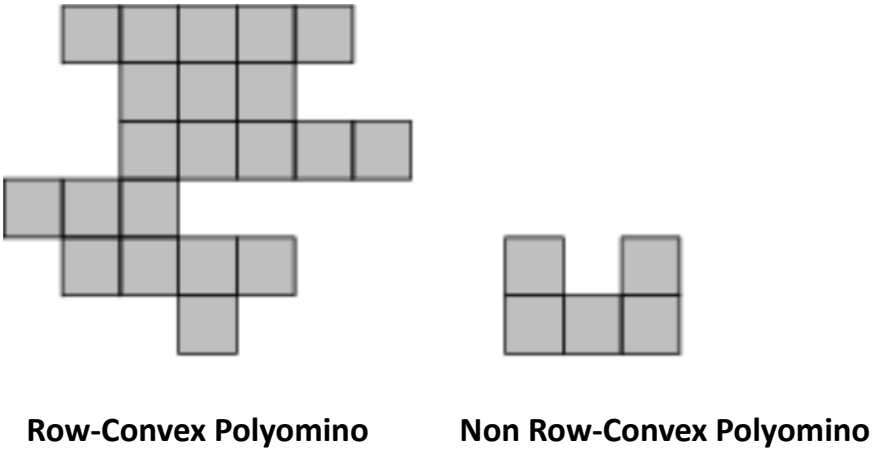


Figure 3.4: Row-convex and non row-convex polyominoes. Image from mathworld.wolfram.com.

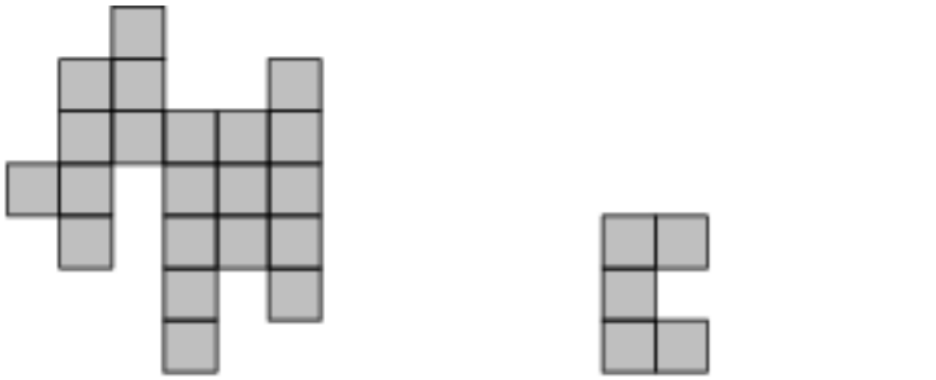


Figure 3.5: Column-convex and non column-convex polyominoes. Image from mathworld.wolfram.com.

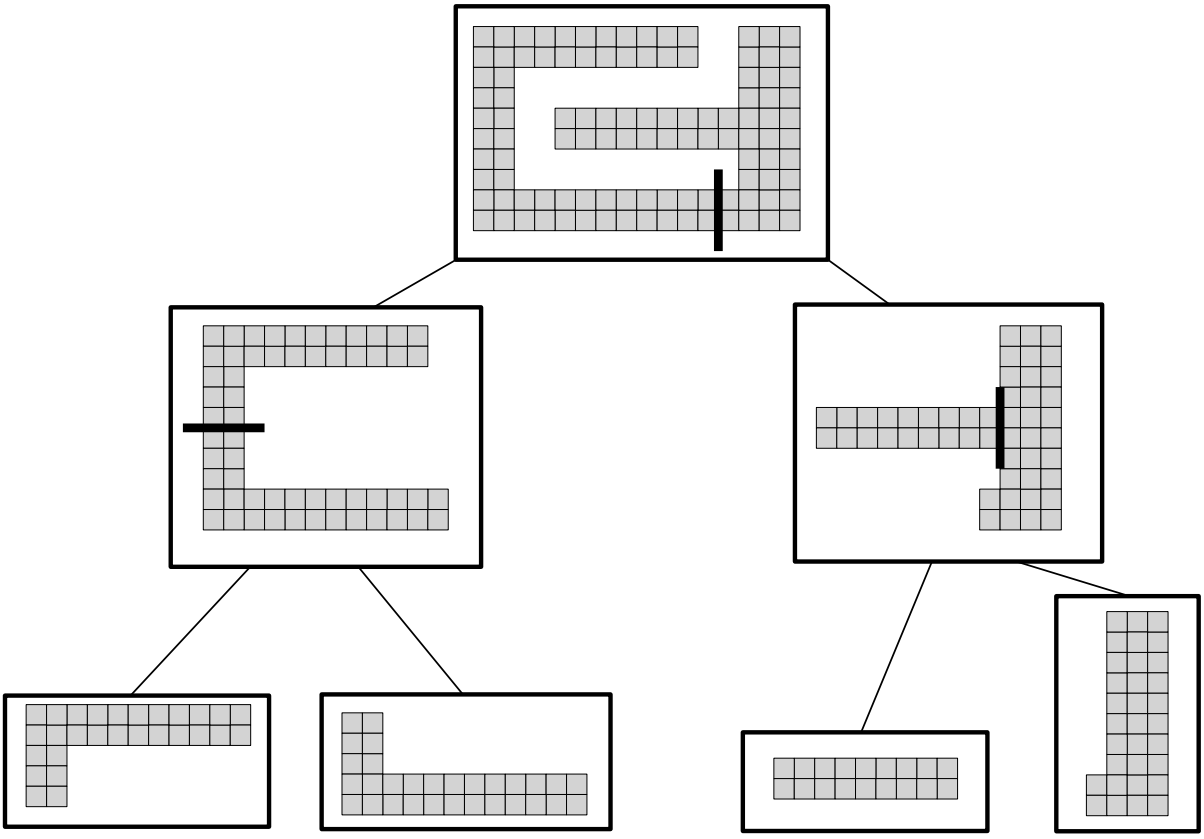


Figure 3.6: Polyomino decomposition using straight cuts.

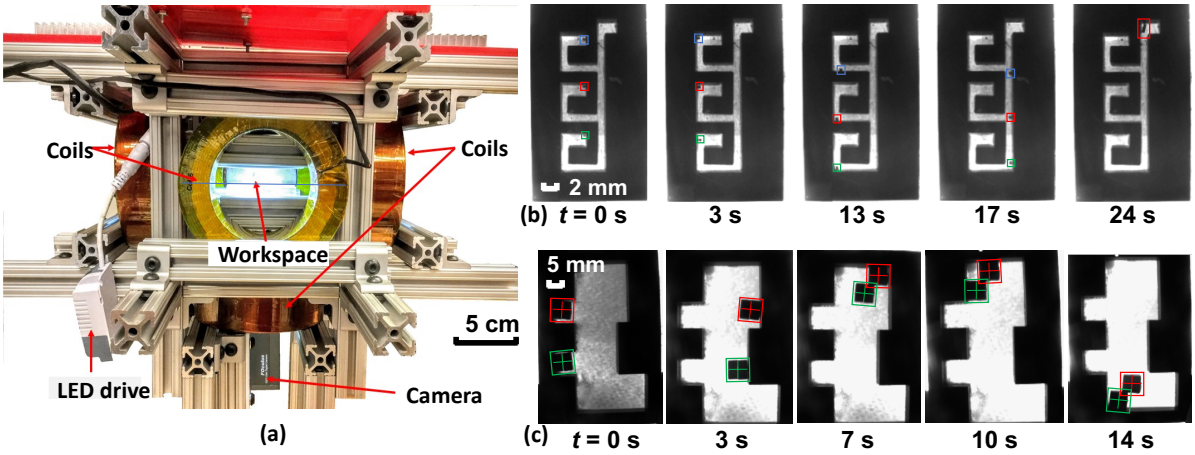


Figure 3.7: (a) Magnetic manipulation workspace (b) frames from an assembly of one column of a polyomino. (c) frames from combining two polyominoes.

workspace. With flux concentration cores, up to 900 mT/m gradient fields are observed in the experiment. A flux concentration core is made of one solid iron cylinder (50.8 mm in diameter), and two outer iron hollow cylinders (respectively 60.7 mm and 73.1 mm in diameter). The workspaces used to demonstrate the sublinear assembly algorithms were designed to replicate the column assembly in Fig. 3.1 and the subpolyomino assembly in Fig. 3.2. Each workspace is made up of two layers of acrylic cut using a Universal Laser Cutter. The base layer is fabricated from 2 mm thick transparent acrylic, and it is glued to a 5.5 mm thick black acrylic, which acts as an obstacle layout. In each experiment, the workspace is filled with detergent, and placed in the center of our electromagnetic platform. The particle tiles are composed of nickel-plated neodymium cube-shaped magnets (supermagnetman.com C0010). The size of the magnet cubes is 0.5 mm^3 for micro scale and 2.88 mm^3 for milli scale demonstration. An Arduino Mega 2560 was used to control the current in the coils and the workspaces were observed with a IEEE 1394 camera, captured at 60 fps. The field of view was 75.7 mm $\times$ 51.7 mm and 37.4 mm $\times$ 37.3 mm for milli-scale and micro-scale respectively.

3.2.2 Experimental Results

In micro-scale experiments, we placed three cubes in each hopper of the workspace. The workspace used in these experiments is 20 mm wide and 39.5 mm long. To assemble the column polyomino, a gradient magnetic field of 900 mT/m was applied in the direction sequence $\langle l, d, r, u \rangle$. Each direction input was applied until all the magnetic particles reached the obstacle layout wall. Fig. 4.3b shows the completed three tile polyomino. For milli-scale experiments we assembled two polyominoes as shown in Fig. 4.3c. Each polyominoes is composed of four cubes glued together to form a square shape. The 43 mm $\times$ 62 mm workspace was placed in a 101 Gauss uniform magnetic field to control the orientation of the polyominoes and then tilted in the direction sequence $\langle u, l, d, r, u \rangle$. See video attachment for experimental demonstrations.

Chapter 4

Sorting of Polyominoes

This chapter is based on the conference paper [45], **'On Designing 2D Discrete Workspaces to Sort or Classify Polyominoes'** by *Phillip Keldenich, Sheryl Manzoor, Li Huang, Dominik Krupke, Arne Schmidt, Aaron T. Becker, and Sándor P. Fekete.* I contributed to hardware experimentation for this work, and the initial proof of concepts for sorting small polyominoes, and the concept of polyomino cams.
Fig. 4.1, Fig. 4.2 and, Fig. 4.4 are from [45].

4.1 Sorting in Static Workspaces

Static workspaces comprise of rigid obstacles and they are simple to design. Polyominoes are sorted one at a time and limited types of shapes can be sorted using static workspaces. For sorting, the boundary of a polyomino is first decomposed into constant and monotonic regions as shown in Fig. 4.4. After decomposition, the polyominoes are sorted as in Fig. 4.2.

4.2 Sorting in Dynamic Workspaces

Dynamic workspaces have moving obstacles which enable them to sort more shapes than static workspaces. The moving obstacles (cams) are affected by the global control input in the same way as the polyominoes being sorted. Fig. 4.1 shows a dynamic 2D workspace with a moving cam that sorts polyominoes based on shape in six moves. A 3×2 polyomino missing the rightmost tile from the middle row exits at the right (red), while a 3×2 polyomino missing the rightmost and middle tile from the middle row exits

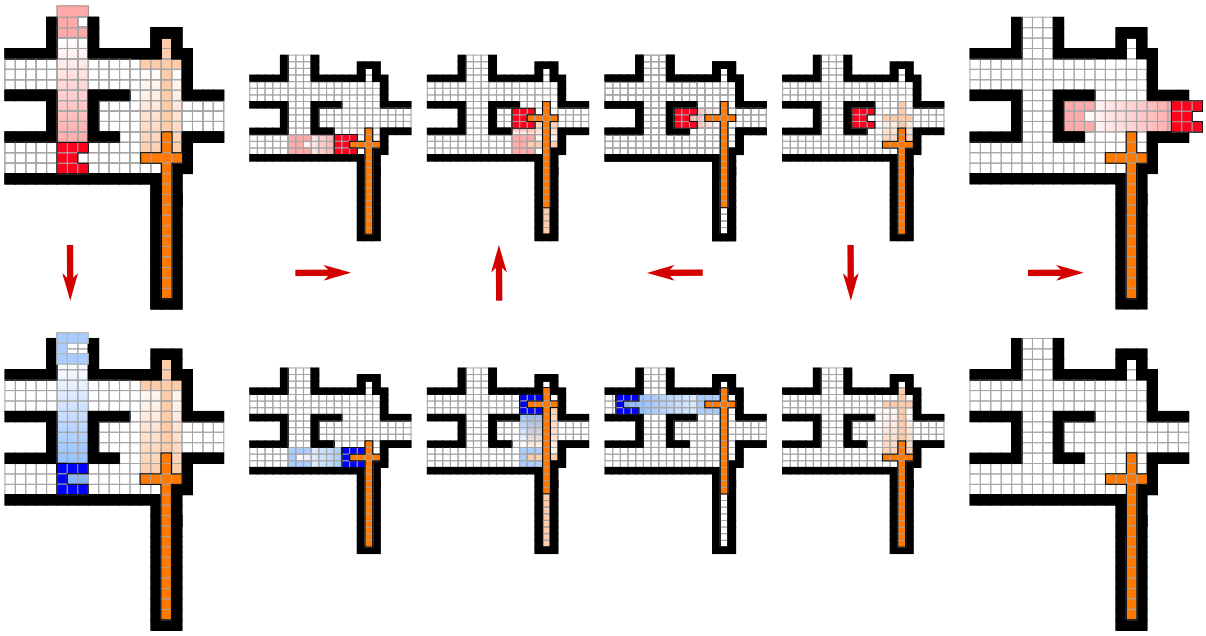


Figure 4.1: A 2D dynamic workspace.

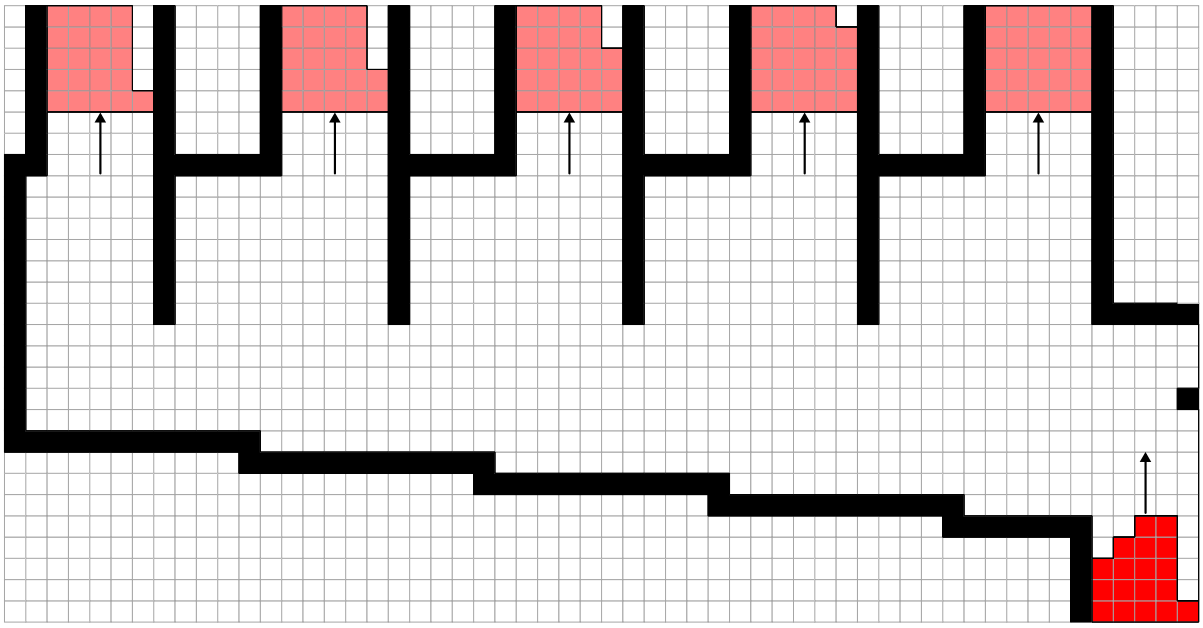


Figure 4.2: Applying the control sequence u, l, u, r, u classifies a polyomino based on the row of the top end of the rightmost constant piece of the boundary.

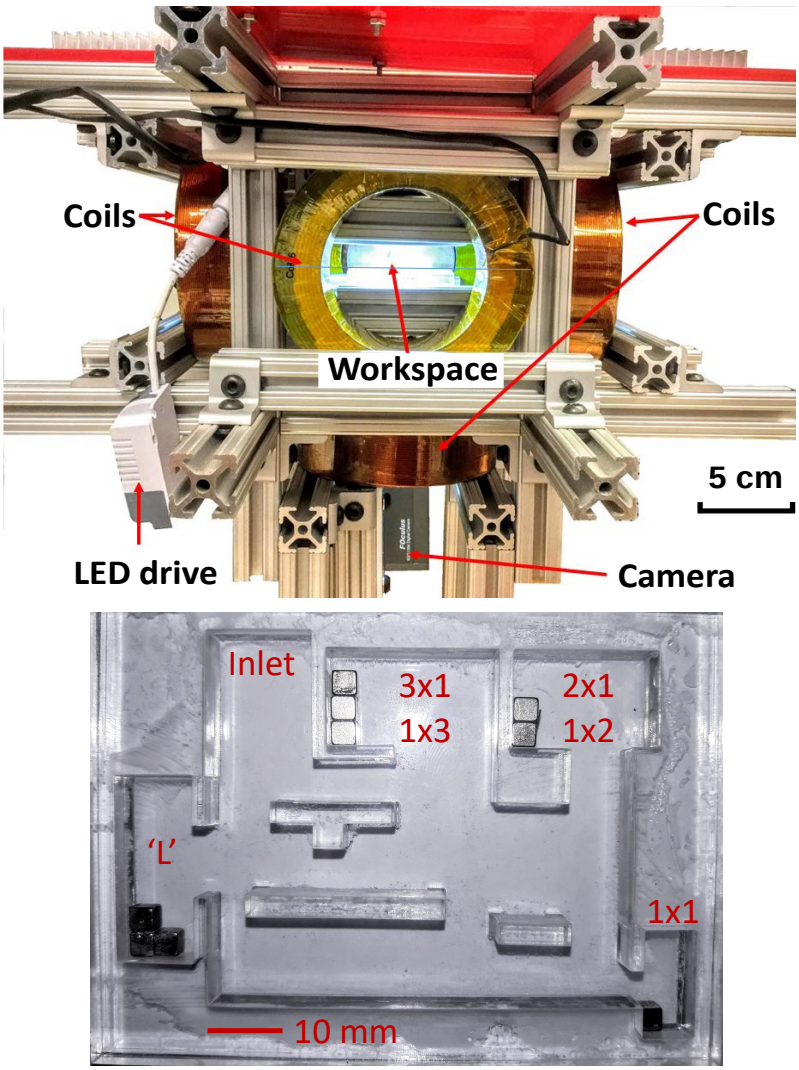


Figure 4.3: (Top) Magnetic manipulation system used to demonstrate polyomino sorting and error detection. (Bottom) 77 mm × 56 mm workspace used to sort all polyominoes with 1, 2, or 3 tiles.

at the top left (red).

4.3 Experimental Demonstration

4.3.1 Experimental Platform

We demonstrated tilt sorting and error detection at the milli-scale using a customized setup that generates a uniform magnetic field to keep the parts aligned in the

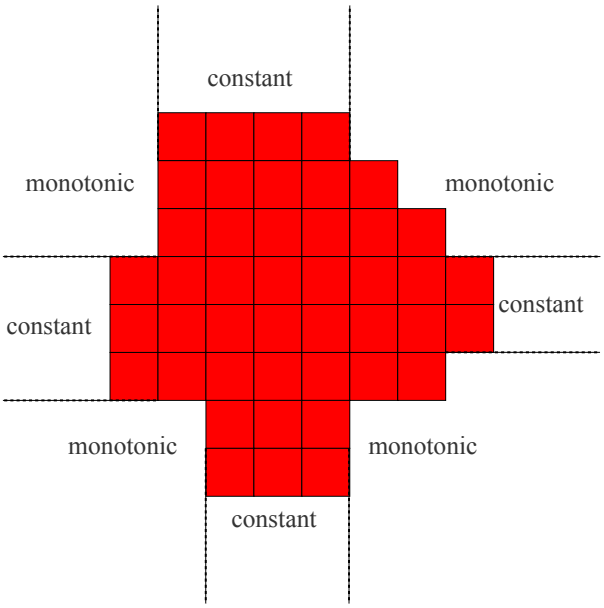


Figure 4.4: Decomposition of the boundary of an orthoconvex polyomino.

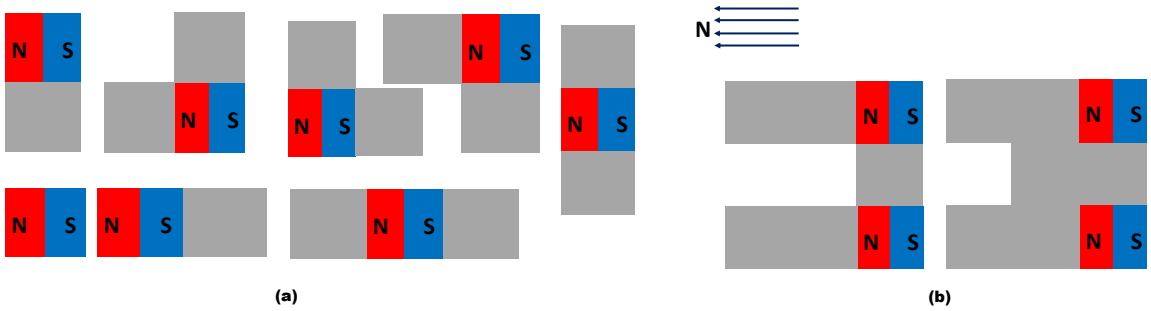


Figure 4.5: (a) Schematic of the eight polyominoes used to demonstrate sorting and their alignment in a uniform magnetic field. (b) Schematic of two polyominoes fabricated for error detection.

commanded direction and gravity is used to manipulate the parts. The customized electromagnetic system in Fig. 4.3 has two pairs of coils (18 AWG, 1200 turns, Custom Coils, Inc) arranged orthogonally and powered by four SyRen10-25 motor drivers. Tekpower HY3020E is used as DC power supply. The system can generate up to 101 Gauss uniform fields on the horizontal plane of the workspace center. The coil current was controlled using an Arduino Mega 2560.

Each workspace used to demonstrate tilt sorting and error detection was designed in AutoCAD and then cut using a Universal Laser Cutter. Two layers of transparent

acrylic were glued together (Gorilla Super Glue) to make a workspace. One layer of 2.0 mm thickness was used as the base and another 5.5 mm thick layer made the obstacle boundary. To perform an experiment, the workspace was placed in the center of the magnetic platform and observed with an IEEE 1394 camera, captured at 60 fps. The polyominoes used for the experiments were fabricated from nickel-plated neodymium cube-shaped magnets (supermagnetman.com C0010 and C0030).

4.3.2 Static Workspace Experiments

To show sorting for static workspaces, we designed two workspaces at two scales. The first system used a workspace of 77 mm width and 56 mm length and sorted polyominoes composed of 3.0 mm^3 neodymium cube magnets. The second, smaller system used a workspace of 44 mm width and 35 mm length and sorted polyominoes made of 1.0 mm^3 neodymium cube magnets. An approximate uniform field of 30 Gauss was employed to keep the polyominoes aligned, and the workspace was tilted in the direction sequence $\{d, r, u, l\}$. The direction inputs were applied until the polyomino touched a layout wall. Fig. 4.5(a) shows the eight polyominoes which were sorted in these experiments. To make a polyomino, one magnetic cube was attached to one or more cubes that had been demagnetized using a blow torch. Fig. 4.6 shows the four different polyomino shapes in their respective bins inside the workspace and the results after applying control sequence for sorting three polyominoes.

4.3.3 Dynamic Workspace Experiments

This experiment used a 59 mm $\times$ 52 mm workspace. A cross-shaped moving cam was used to detect the inner shape of two polyominoes. One polyomino had a dent one-tile deep while the other had a dent two-tiles deep. The cam and the workspace in Fig. 4.7 are designed so that the polyomino with a one-tile dent is stored in a bin and the other polyomino is rejected. The direction sequence for the parts and the cam

is $\{d, r, u, l\}$. Each of the two polyominoes contains two magnetic cubes, one in the top row and the other in the bottom, attached to the demagnetized cubes as shown in Fig. 4.5(b). Fig. 4.7 (Top) shows a dynamic workspace with one sliding cam, designed for error detection using the sequence $\{d, r, u, l\}$. Only 3×3 polyominoes with a one-tile dent on the right side, are moved into the storage area; all others are eliminated and exit the sorter. Fig. 4.7 (Bottom) shows results for error detection with two polyomino shapes, a two-tile dent (top) and a one-tile dent (bottom).

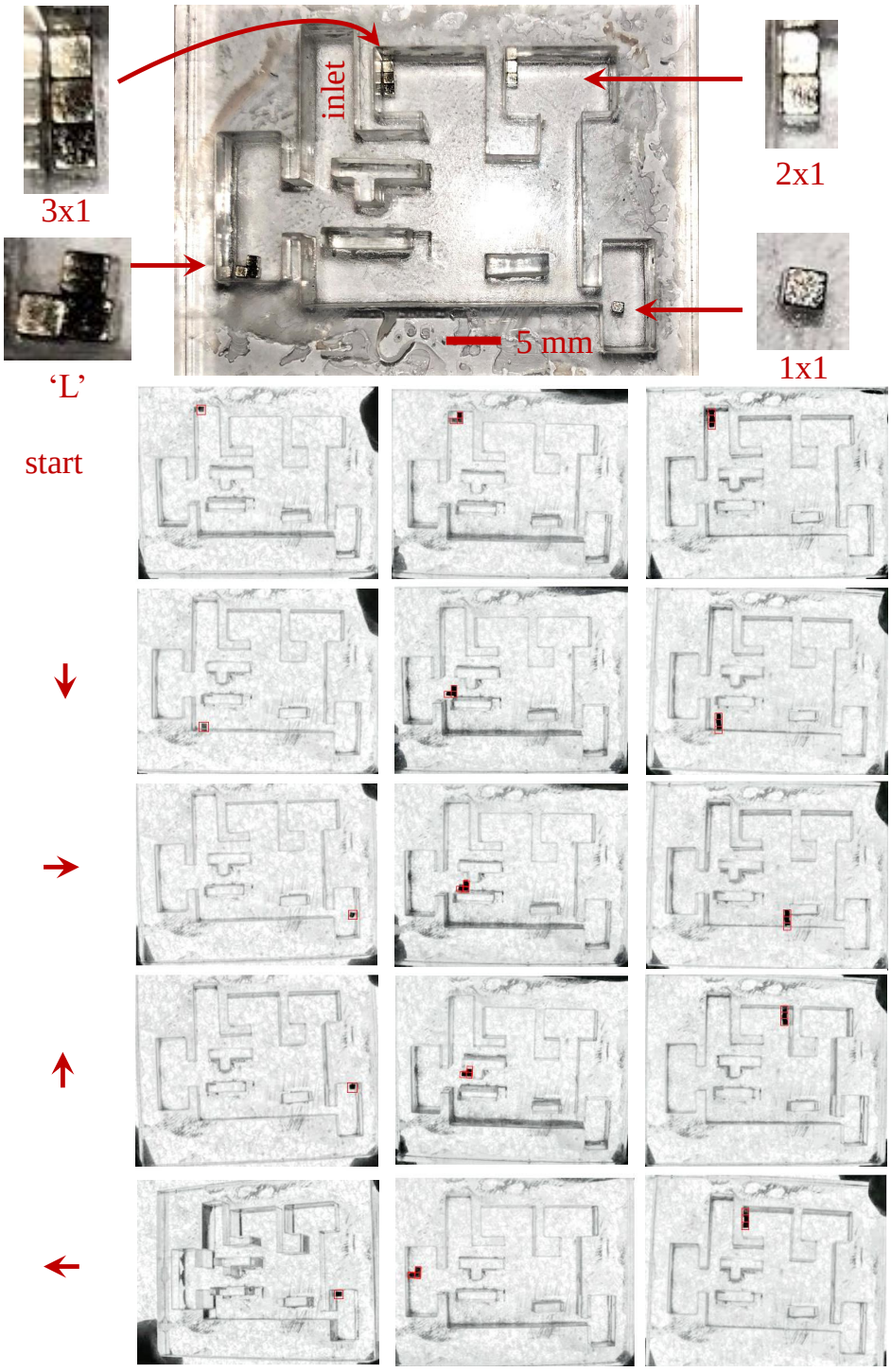


Figure 4.6: Frames from video demonstration of sorting polyominoes using the static workspace for 1 mm tiles.

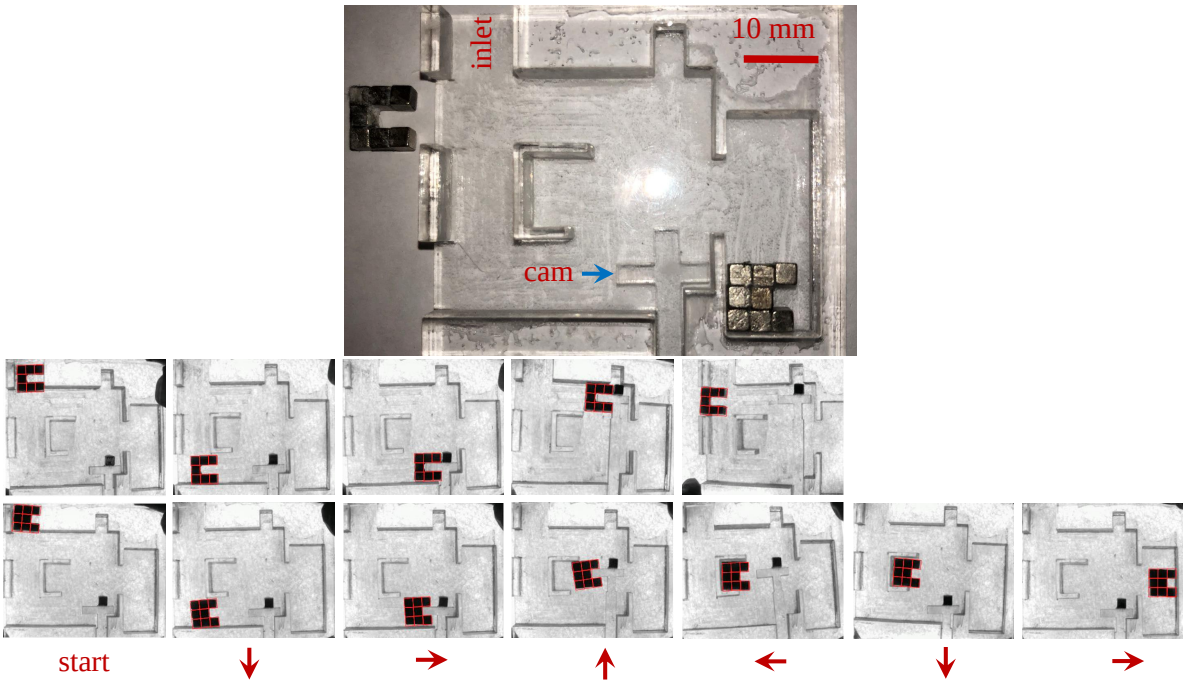


Figure 4.7: The polyomino with a one-tile dent is stored inside an output region and the other polyomino exits the workspace after applying the control input sequence $\{d, r, u, l\}$.

Chapter 5

Conclusion & Future Work

This work introduces a new model for additive assembly that enables efficient parallel construction of arbitrary 2D shapes and it does not depend on individual control of each agent or particle. This enables a simple global control input to produce a complex output. The algorithms presented can be used to automatically generate a factory layout which can produce the desired copies of the required shape when it is actuated by the global control input.

The thesis begins with a discussion of sequential parallel self-assembly of polyominoes where one particle is added at a time to build a complete shape. The algorithms for finding the build sequence of the shape and construction of the factory layout is detailed in Chapter 2. Additionally, the chapter also discusses hardware experimental results for the assembly algorithms. Future work could extend Algorithms 1–5 to three dimensions and increase the robustness of micro- and macro-scale assembly. Furthermore, techniques to improve particle movement speed should be investigated and different particle types should be explored and fabricated to implement the assembly of polyominoes.

In Chapter 3, a more efficient approach for assembling polyominoes is presented. It is efficient because it requires less number of cycles of control input sequence to build a shape. Parts are decomposed into subassemblies, which enables more complex parts to be created and it also enables construction in logarithmic number of cycles. Finally, algorithms to sort polyominoes are discussed in Chapter 4. After the polyominoes have been assembled, they are sorted to check for any errors in the shapes. Future research directions could focus on determining the shapes which cannot be decomposed using straight cuts and then devise algorithms for their assembly.

References

[1] B. R. Donald, C. G. Levey, I. Paprotny, and D. Rus, "Planning and control for microassembly of structures composed of stress-engineered MEMS microrobots," *The International Journal of Robotics Research*, vol. 32, no. 2, pp. 218–246, 2013. [Online]. Available: http://ijr.sagepub.com/content/32/2/218.abstract

[2] P.-T. Chiang, J. Mielke, J. Godoy, J. M. Guerrero, L. B. Alemany, C. J. Villagómez, A. Saywell, L. Grill, and J. M. Tour, "Toward a light-driven motorized nanocar: Synthesis and initial imaging of single molecules," *ACS Nano*, vol. 6, no. 1, pp. 592–597, Feb. 2011.

[3] H.-W. Tung, D. R. Frutiger, S. Panè, and B. J. Nelson, "Polymer-based wireless resonant magnetic microrobots," in *IEEE International Conference on Robotics and Automation*, May 2012, pp. 715–720.

[4] E. Diller, J. Giltinan, and M. Sitti, "Independent control of multiple magnetic microrobots in three dimensions," *The International Journal of Robotics Research*, vol. 32, no. 5, pp. 614–631, 2013. [Online]. Available: http://ijr.sagepub.com/content/32/5/614.abstract

[5] W. Jing, N. Pagano, and D. Cappelleri, "A tumbling magnetic microrobot with flexible operating modes," in *IEEE International Conference on Robotics and Automation*, May 2013, pp. 5514–5519.

[6] Y. Ou, D. H. Kim, P. Kim, M. J. Kim, and A. A. Julius, "Motion control of magnetized tetrahymena pyriformis cells by magnetic field with model predictive control," *Int. J. Rob. Res.*, vol. 32, no. 1, pp. 129–139, Jan. 2013.

[7] D. de Lanauze, O. Felfoul, J.-P. Turcot, M. Mohammadi, and S. Martel, "Three-dimensional remote aggregation and steering of magnetotactic bacteria microrobots for drug delivery applications," *The International Journal of Robotics Research*, 11 2013. [Online]. Available: http://ijr.sagepub.com/content/early/2013/11/11/0278364913500543

[8] A. Becker, E. Demaine, S. Fekete, G. Habibi, and J. McLurkin, "Reconfiguring massive particle swarms with limited, global control," in *International Symposium on Algorithms and Experiments for Sensor Systems, Wireless Networks and Distributed Robotics (ALGOSENSORS)*, Sophia Antipolis, France, Sep. 2013, pp. 51–66.

[9] A. Becker, E. Demaine, S. Fekete, and J. McLurkin, "Particle computation: Designing worlds to control robot swarms with only global signals," in *IEEE International Conference on Robotics and Automation (ICRA)*. Hong Kong: IEEE, May 2014, pp. 6751–6756.

[10] A. Becker, E. D. Demaine, S. P. Fekete, G. Habibi, and J. McLurkin, "Reconfiguring massive particle swarms with limited, global control," in *Algorithms for Sensor Systems*, ser. Lecture Notes in Computer Science, P. Flocchini, J. Gao, E. Kranakis, and F. Meyer auf der Heide, Eds. Springer Berlin Heidelberg, 2014, vol. 8243, pp. 51–66. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-45346-5_5

[11] T. M. S. Chang, "Therapeutic applications of polymeric artificial cells," *Nature Reviews Drug Discovery*, vol. 4, no. 3, pp. 221–235, 2005.

[12] S. Prakash, *Artificial cells, cell engineering and therapy*. Cambridge, U.K.: Woodhead Publishing Limited, and Boca Raton, FL, USA: CRC Press, 2007.

[13] T. M. S. Chang, *Artificial cells: biotechnology, nanomedicine, regenerative medicine, blood substitutes, bioencapsulation, and cell/stem cell therapy*. World Scientific, 2007, vol. 1.

[14] D. B. Weibel, W. R. DiLuzio, and G. M. Whitesides, "Microfabrication meets microbiology," *Nature Reviews Microbiology*, vol. 5, no. 3, pp. 209–218, 2007.

[15] J. J. Abbott, Z. Nagy, F. Beyeler, and B. Nelson, "Robotics in the small," *IEEE Robotics & Automation Magazine*, vol. 14, no. 2, pp. 92–103, 2007.

[16] C. Yi, C.-W. Li, S. Ji, and M. Yang, "Microfluidics technology for manipulation and analysis of biological cells," *Analytica Chimica Acta*, vol. 560, no. 1, pp. 1–23, 2006.

[17] J. Castillo, M. Dimaki, and W. E. Svendsen, "Manipulation of biological samples using micro and nano techniques," *Integrative Biology*, vol. 1, no. 1, pp. 30–42, 2009.

[18] M. Sitti, H. Ceylan, W. Hu, J. Giltinan, M. Turan, S. Yim, and E. Diller, "Biomedical applications of untethered mobile milli/microrobots," *Proceedings of the IEEE*, vol. 103, no. 2, pp. 205–224, Feb 2015.

[19] R. E. Assal, P. Chen, and U. Demirci, "Highlights from the latest articles in advanced biomanufacturing at micro-and nano-scale," *Nanomedicine*, vol. 10, no. 3, pp. 347–350, 2015.

[20] J. P. Desai, A. Pillarisetti, and A. D. Brooks, "Engineering approaches to biomanipulation," *Annu. Rev. Biomed. Eng.*, vol. 9, pp. 35–53, 2007.

[21] P. Y. Chiou, A. T. Ohta, and M. C. Wu, "Massively parallel manipulation of single cells and microparticles using optical images," *Nature*, vol. 436, no. 7049, pp. 370–372, 2005.

[22] C. W. Shields IV, C. D. Reyes, and G. P. López, "Microfluidic cell sorting: a review of the advances in the separation of cells from debulking to rare cell isolation," *Lab on a Chip*, vol. 15, no. 5, pp. 1230–1249, 2015.

[23] P. Augustsson, J. Persson, S. Ekström, M. Ohlin, and T. Laurell, "Decomplexing biofluids using microchip based acoustophoresis," *Lab on a Chip*, vol. 9, no. 6, pp. 810–818, 2009.

[24] D. Vigolo, R. Rusconi, H. A. Stone, and R. Piazza, "Thermophoresis: microfluidics characterization and separation," *Soft Matter*, vol. 6, no. 15, pp. 3489–3493, 2010.

[25] Y.-H. Lin, Y.-W. Yang, Y.-D. Chen, S.-S. Wang, Y.-H. Chang, and M.-H. Wu, "The application of an optically switched dielectrophoretic (odep) force for the manipulation and assembly of cell-encapsulating alginate microbeads in a microfluidic perfusion cell culture system for bottom-up tissue engineering," *Lab on a Chip*, vol. 12, no. 6, pp. 1164–1173, 2012.

[26] M. Rubenstein, C. Ahler, and R. Nagpal, "Kilobot: A low cost scalable robot system for collective behaviors," in *IEEE International Conference on Robotics and Automation*. IEEE, 2012, pp. 3293–3298.

[27] Y. Ou, D. H. Kim, P. Kim, M. J. Kim, and A. A. Julius, "Motion control of magnetized tetrahymena pyriformis cells by a magnetic field with model predictive control," *The International Journal of Robotics Research*, vol. 32, no. 1, pp. 129–140, 2013.

[28] P.-T. Chiang, J. Mielke, J. Godoy, J. M. Guerrero, L. B. Alemany, C. J. Villagomez, A. Saywell, L. Grill, and J. M. Tour, "Toward a light-driven motorized nanocar: Synthesis and initial imaging of single molecules," *ACS nano*, vol. 6, no. 1, pp. 592–597, 2011.

[29] S. Chowdhury, W. Jing, and D. J. Cappelleri, "Controlling multiple microrobots: recent progress and future challenges," *Journal of Micro-Bio Robotics*, vol. 10, no. 1-4, pp. 1–11, 2015.

[30] B. R. Donald, C. G. Levey, I. Paprotny, and D. Rus, "Planning and control for microassembly of structures composed of stress-engineered mems microrobots," *The International journal of robotics research*, vol. 32, no. 2, pp. 218–246, 2013.

[31] G. Mermoud, M. Mastrangeli, U. Upadhyay, and A. Martinoli, "Real-time automated modeling and control of self-assembling systems," in *2012 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2012, pp. 4266–4273.

[32] M. Mastrangeli, F. Schill, J. Goldowsky, H. Knapp, J. Brugger, and A. Martinoli, "Automated real-time control of fluidic self-assembly of microparticles," in *Robotics and Automation (ICRA), 2014 IEEE International Conference on*. IEEE, 2014, pp. 5860–5865.

[33] A. Becker, G. Habibi, J. Werfel, M. Rubenstein, and J. McLurkin, "Massive uniform manipulation: Controlling large populations of simple robots with a common input signal," in *IEEE/RSJ International Conf. on Intelligent Robots and Systems*. IEEE, 2013, pp. 520–527.

[34] S. Tasoglu, E. Diller, S. Guven, M. Sitti, and U. Demirci, "Untethered micro-robotic coding of three-dimensional material composition," *Nature communications*, vol. 5, 2014.

[35] S. Tasoglu, C. Yu, H. Gungordu, S. Guven, T. Vural, and U. Demirci, "Guided and magnetic self-assembly of tunable magnetoceptive gels," *Nature communications*, vol. 5, 2014.

[36] J. Seymour and D. J. Cappelleri, "Automated microassembly sequence planning with sub-assemblies," in *ASME 2016 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*. American Society of Mechanical Engineers, 2016, pp. V004T08A018–V004T08A018.

[37] M. F. F. Rashid, W. Hutabarat, and A. Tiwari, "A review on assembly sequence planning and assembly line balancing optimisation using soft computing approaches," *The International Journal of Advanced Manufacturing Technology*, vol. 59, no. 1, pp. 335–349, 2012.

[38] Q. Su, "Computer aided geometric feasible assembly sequence planning and optimizing," *The International Journal of Advanced Manufacturing Technology*, vol. 33, no. 1, pp. 48–57, 2007. [Online]. Available: http://dx.doi.org/10.1007/s00170-006-0447-0

[39] L. Mertz, "Tiny conveyance: Micro- and nanorobots prepare to advance medicine," *IEEE Pulse*, vol. 9, no. 1, pp. 19–23, Jan 2018.

[40] A. T. Becker, S. P. Fekete, P. Keldenich, D. Krupke, C. Rieck, C. Scheffer, and A. Schmidt, "Tilt assembly: Algorithms for micro-factories that build objects with uniform external forces," in *LIPIcs-Leibniz International Proceedings in Informatics*, vol. 92. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2017.

[41] S. Manzoor, S. Sheckman, J. Lonsford, H. Kim, M. J. Kim, and A. T. Becker, "Parallel self-assembly of polyominoes under uniform control inputs," *IEEE Robotics and Automation Letters*, vol. 2, no. 4, pp. 2040–2047, Oct 2017.

[42] A. Schmidt, personal communication.

[43] S. Manzoor and A. T. Becker, "Github link for particle assembly code, https://github.com/aabecker/particlecomputation/tree/master/assembly," Feb. 2017.

[44] A. Schmidt, S. Manzoor, L. Huang, A. T. Becker, and S. P. Fekete, "Efficient parallel self-assembly under uniform control inputs," *IEEE Robotics and Automation Letters*, vol. 3, no. 4, pp. 3521–3528, Oct 2018.

[45] P. Keldenich, S. Manzoor, L. Huang, D. M. Krupke, A. Schmidt, S. P. Fekete, and A. T. Becker, "On designing 2d discrete workspaces to sort or classify 2d polyominoes," in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018.