



Technische
Universität
Braunschweig

BACHELOR THESIS

Motion Planning for Reconfigurable Magnetic Modular Cubes in the 2-Dimensional Special Euclidean Group

Kjell Keune

Institut für Betriebssysteme und Rechnerverbund

Supervised by
Prof. Dr. Aaron T. Becker

March 29, 2023

Statement of Originality

This thesis has been performed independently with the support of my supervisor/s. To the best of the author's knowledge, this thesis contains no material previously published or written by another person except where due reference is made in the text.

Braunschweig, March 29, 2023

Aufgabenstellung / Task Description

Deutsch: Um spezifische Aufgaben besser zu bewältigen, lassen sich modulare, rekonfigurierbare Roboter zu größeren Strukturen zusammensetzen und wieder auseinandernehmen. Magnetic-modular-cubes sind skalierbare Einheiten, bei welchen Permanentmagneten in einen würfelförmigen Körper eingebettet sind. Diese Einheiten zählen als rekonfigurierbare Roboter, obwohl sie selber keine Logik oder Stromversorgung beinhalten. Stattdessen lassen sich diese durch ein externes, gleichmäßiges und sich zeitlich änderndes Magnetfeld steuern. Durch diese Steuerung können die magnetic-cubes auf der Stelle gedreht oder durch pivot-walking nach rechts und links bewegt werden. Obwohl sich das Magnetfeld auf alle Einheiten gleichermaßen auswirkt, kann durch Kollision mit der Arbeitsflächenbegrenzung eine Änderung der Anordnung bewirkt werden. Befinden sich zwei magnetic-cubes nah genug beieinander können sich diese durch die Permanentmagneten miteinander verbinden und so Polyominos als größere Strukturen aufbauen, welche auf die gleiche Weise wie einzelne cubes gesteuert werden können. Frühere Arbeiten betrachteten das "tilt-model", bei welchem sich Strukturen jeder Größe mit gleicher Geschwindigkeit in ganzzahligen Schritten und mit ausschließen 90° Drehungen bewegen lassen.

Herr Keunes Aufgabe in dieser Bachelorarbeit ist es, einen motion-planner für die beschriebenen magnetic-cubes zu entwerfen, welcher mit beliebigen Positionen und Rotationen umgehen kann. Dabei ist es erforderlich, eine Simulationsumgebung zu schaffen, welche das Verhalten der magnetic-cubes repliziert. Es soll ein lokaler motion-planner entwickelt werden, um zwei Polyominos an gewünschten Kanten zu verbinden. Dieser local-planner soll Heuristiken und optimale Bewegungsabläufe mit möglichst wenig Schritten realisieren. Ebenfalls soll dieser global eingesetzt werden, um Bewegungsabläufe zu finden, die gewünschte Polyominos aus einer zufällig gegebenen Startkonfiguration erzeugen. Ein interessantes Ergebnis wird es sein, zu sehen, wie gut Probleminstanzen dieser Art in der Realität gelöst werden können und welche Parameter die gravierendsten Auswirkungen auf die Schwierigkeit von motion-planning Problemen haben.

English: Reconfigurable modular robots can dynamically assemble/disassemble to better accomplish a desired task. Magnetic modular cubes are scalable modular subunits with embedded permanent magnets in a 3D-printed cubic body. These cubes can act as reconfigurable modular robots, even though they contain no power, actuation or computing. Instead, these cubes can be wirelessly controlled by an external, uniform, time-varying magnetic field. This control allows the cubes to spin in place or pivot walk to the left or right direction. Although the applied magnetic field is the same for each magnetic modular cube, collisions with workspace boundaries can be used to rearrange the cubes. Moreover, the cubes magnetically self-assemble when brought in close proximity of another cube, and form polyominoes, which can be controlled the same way as single cubes. Related work has considered the “tilt model,” where similar cubes and polyominoes move between integer positions, all move at the same speed, and only rotate by 90 degree steps.

In his thesis, Mr. Keune’s task is to design a motion planner for magnetic cubes that can assume arbitrary positions and orientations in the workspace. This requires designing a simulation environment that replicates the behavior of magnetic cubes. He will design local planners for moving two polyominoes to assemble at desired faces. Designing the local planner includes heuristics and computing optimal motion plans that minimize the number of steps. The local planner will be used to search for global planning sequences to generate desired polyominoes from a given starting configuration. One exciting outcome will be studying how well instances can be solved in practice and analyzing which parameters have the most significant effect on the difficulty of the motion planning problem.

Abstract

Abstract

Contents

1	Introduction	1
1.1	Related Work	1
1.2	Contribution	3
2	Preliminaries	5
2.1	Magnetic Modular Cubes	6
2.2	Workspace And Configuration	7
2.3	Polyominoes	8
2.4	Motion Modes	9
3	Local Planner	13
3.1	Connecting Polyominoes	13
3.2	Aligning Cubes	14
3.3	Walking And Waiting	14
3.4	Plan And Plan-State	14
3.5	Local Planning Algorithm	14
4	Global Planner	15
5	Simulator	17

List of Figures

2.1	Simplified top-down view of the two magnetic modular cube types with their outward pointing magnet poles, illustrated as red and blue squares. Also visualizes the lengths r_C and r_M	5
2.2	Rectangular workspace with a configuration of four magnetic modular cubes. All cubes have the same orientation as the magnetic field, indicated by the compass in the top-left corner.	7
2.3	Examples of polyominoes and their equality. a) and d) are equal, only the magnetic field changed its orientation. a) and c) are not equal, they have the same shape but rotated. a) and c) are also not equal because of different cube types in the same shape. e) shows an invalid polyomino in its grid representation (top) and how it behaves in the simulation (bottom).	8
2.4	This Figure describes the pivot walking motion in detail. a) shows the six pivot walking steps for a single red cube. You can see the orientation of the magnetic field (bigger arrow indicates elevation). In b) an example polyomino with its pivot axis, edges and points is shown. c) illustrates the rotation of the pivot axis labeled with all the pivot walking parameters.	9
2.5	All 19 four-cube polyomino shapes with their displacement vector \vec{d} for one pivot walking cycle with $\alpha = \frac{\pi}{4}$. \vec{d} is drawn from the center of mass (red dot). North and south pivot point are drawn as blue and brown dots.	10
3.1	Tree different examples for connecting polyominoes into holes. a) and b) show one-cube-deep holes (a) east-west and b) north-south). c) illustrates a two-cube-deep east-west hole. The edges to be connected are marked yellow.	14

1 Introduction

Self-assembling modular parts forming bigger structures is a well-known concept in nature and most functionalities of living organisms follow this principle [5]. DNA for example has the ability to self-replicate by using differently shaped proteins, that combine themselves in various ways. At larger sizes, these cells can be combined to assemble things like tissue, organs and even whole organisms. Complex structures can be assembled and disassembled depending on the task they should accomplish at a given point in time. Using self-reconfiguring robot swarms in such a way has promising applications in the future. Biomedical applications could be targeted drug delivery or drug screening [15], or a robot swarm could be used for milliscale and microscale manufacturing [12].

Designing robots at these small sizes faces challenging problems. Equipping each robot with its own sensors, actuation-system, connection-system and power supply seem infeasible, in terms of the miniaturization required and power-limitations [16]. Therefore, the use of external global control, effecting every robot in the workspace with the same torque and force, is a promising solution [16]. Using robots with embedded permanent magnets, has all the desired effects. Robots can be controlled by an external magnetic field and also connect to each other without any internal power supply and for sensing an external camera can be used [13].

One example for magnetically controlled robots are the magnetic modular cubes by Bhattacharjee et al. [4], which are the subjects of this thesis. We will develop a simulation that simulates the behavior of magnetic modular cubes, without assuming discrete movement or limiting rotations to a certain amount. The simulation will be used for developing closed-loop planning algorithms, which provide a control sequence to assemble desired target shapes. For that it is necessary to develop a local planner that is able to connect structures at desired faces. We will look at the difficulties and problems that occur, when working with magnetic modular cubes in the 2-dimensional special Euclidean group $SE(2)$, the space of rigid movements in a 2-dimensional plane.

1.1 Related Work

Continuous motion planning is a crucial subject in the field of robotics. The goal is to find a path from the initial state of a robot to a desired goal state, by performing actions which the robot is capable of. The movement may result in collision with static obstacles and with other robots. The state of the system is also called a configuration. All possible configurations one or multiple robots can be in is defined as the configuration-space. Increasing the dimension of the configuration-space gains rapidly in complexity by

1 Introduction

increasing the number of robots and possible actions. It is difficult to engineer algorithms that explore these huge configuration-spaces and provide continuous path from the initial to the goal configuration, or report failure, if the goal is not reachable. A lot of research was done on motion planning and the textbooks [6] and [10] offer a great overview and also explain a lot of important concepts in detail.

When working with configuration-spaces that are uncountable infinite, like the special Euclidean group, one concept that has been successful for many robotics problem is sample-based motion planning. By taking samples, you can reduce the planning problem from navigating a configuration space to planning on a graph, but you might lose possible solutions. Algorithms like that are not complete anymore, but by using a good sampling technique you can get arbitrarily close to any point, and therefore these algorithms can be called resolution complete. Ways of sampling include random sampling or using a grid with a resolution that is dynamically adjustable. After sampling, conventional discrete planning algorithms can be applied [6].

One state-of-the-art sampling-based approach are algorithms that use rapidly-exploring random trees (RRT). This method tries to grow a tree-shaped graph in the configuration space by moving into the direction of randomly chosen samples from already explored configuration. That way the space gets explored uniformly without being too fixated on the goal configuration [7, 8].

When working with multiple robots, the interaction of robots with each other becomes important. One interesting idea is that single robots can connect to form bigger structures. This is referred to as self-assembly and E. Winfree [17] proposed the abstract Tile Assembly Model (aTAM) in the context of assembling DNA. In this model, particles can have different sets of glues and connect according to certain rules regarding the glue type. However, he considers this process as nondeterministic, so there is no exact instruction on how to assemble a desired structure.

One model more related to the magnetic modular cubes used in this thesis is the Tilt model from Becker et al. [1]. In the Tilt model, all tiles move into one of the cardinal directions until hitting an obstacle. Different variations of the model include moving everything only one step, or the maximally possible amount. It offers a solution when robots are controlled uniformly by external global control inputs. In [1] it is shown that transforming one configuration into another, known as the reconfiguration-problem, is NP-hard. Following work [2] also proves that finding an optimal control sequence, minimizing the number of actions, for the configuration-problem is PSPACE-complete. Furthermore, research is done on designing environments in which the Tilt model can be used to accomplish certain tasks. In particular, Becker et al. [2] create connected logic gates that can evaluate logical expressions.

More on the side of self-assembly, in [3] the construction of desired shapes using the tilt model is researched. It presents a method that can determine a building sequence for a polyomino by adding one tile at a time, considering the rules of Tilt. Also examined are ways of modifying the environment to create factories that construct shapes in a pipeline by repeating the same global control inputs. Shapes can be constructed more efficiently by combining multi-tiled shapes to an even bigger structure. One article considering the

construction with so-called sub-assemblies is proposed by A. Schmidt [14].

Most recently, Bhattacharjee et al. [4] developed the magnetic modular cubes. These robots contain embedded permanent magnets and have no computation or power supply. Instead, they are controlled by an external time-varying magnetic field and are able to perform various actions. Most importantly, they can rotate in place or use a technique called pivot walking to move either left or right. The magnets also act as glues and allow the cubes to perform self-assembly. Although it is theoretically possible to assemble 3-dimensional structures, most research was done by only connecting cubes in two dimensions. Since all cubes are the same size, the assembled 2-dimensional shapes can be represented as polyominoes. An enumeration was done on the amount of possible polyominoes that can be created by cubes with different magnet configurations [9].

By limiting the controls to only 90 degree turns and assuming a uniform pivot walking distance for all structures per step, magnetic modular cubes follow rules similar to the Tilt model. Following these limitations, a simple discrete motion planner was developed, that explores a finite configuration-space and lists all the possible polyominoes that can be created from an initial configuration [4]. One interesting paper from Blumenberg et al. [11] explores the assembly of polyominoes in arbitrary environments, when cubes obey the tilt model. He provides different algorithmic approaches using various distance heuristics and even a solution making use of RRTs. For that he follows the rules of Tilt in a discrete setting.

1.2 Contribution

2 Preliminaries

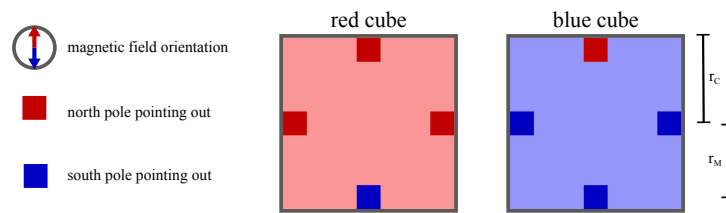


Figure 2.1: Simplified top-down view of the two magnetic modular cube types with their outward pointing magnet poles, illustrated as red and blue squares. Also visualizes the lengths r_C and r_M

2.1 Magnetic Modular Cubes

The magnetic modular cubes are cube-shaped bodies embedded with permanent magnets on the four side faces. The magnets have different orientations of their north and south pole. One pole is always pointing outside and the other straight to the center of the cube. The magnet at the front face has its north pole pointing outwards and the magnet at the back its south pole. These two magnets ensure that the cube is always aligned with the global magnetic field and this orientation holds true for both cube types. The two other side faces must have the same outwards pointing pole, so that it is not possible for this axis to align with the magnetic field.

In fact, this is the reason a distinct definition of front, back and side is even possible. Since the front is always pointing to the north pole of the magnetic field, we also call it the north face, or north edge in two dimensions, and all the other faces can also be called by their corresponding cardinal direction. For simplification we call magnets by their outwards pointing pole in further sections.

Furthermore, two different cube types are defined: Either both side magnets point out their north pole, these cubes are called red cubes, or they point out their south pole, which is then called a blue cube. Figure 2.1 shows a top-down view of the two cube types with all the outwards pointing magnet poles. A compass always shows the orientation of the magnetic field in our illustrations.

Magnetic Modular Cubes can be constructed in different sizes and ways. For more technical details and length measurements, we refer to the original [4]. Two important lengths that we use for planning and simulating are the cube radius r_C and the magnet radius r_M (also illustrated in Figure 2.1). r_C is one half-length of a cube face and r_M is the distance from the center of the cube to the center of the magnet.

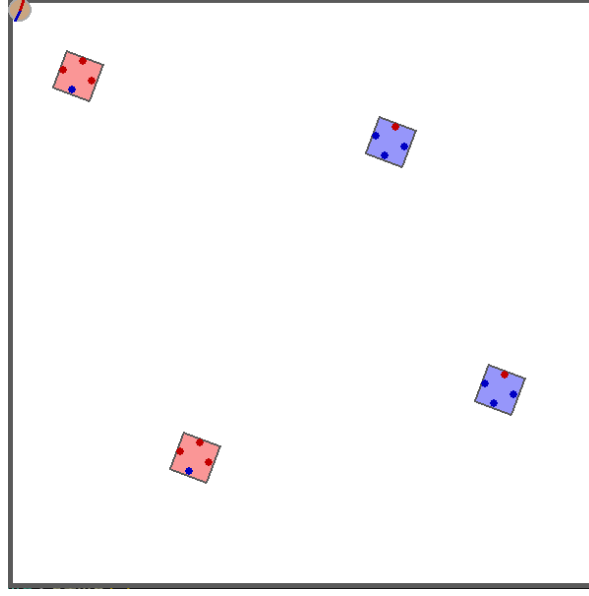


Figure 2.2: Rectangular workspace with a configuration of four magnetic modular cubes. All cubes have the same orientation as the magnetic field, indicated by the compass in the top-left corner.

2.2 Workspace And Configuration

Magnetic modular cubes could theoretical be placed and maneuvered on any 2-dimensional plane with numerous obstacles, as long as you can surround the workspace with a time varying magnetic field. The magnetic field should be able to point in any direction specified by angles of latitude and longitude, so that the cubes can operate in all desired motion modes. Because the motion planning problem of self-assembling target shapes in the special Euclidean group is hard enough without considering obstacles and arbitrary workspace shapes, we only work in a rectangular workspace with no internal obstacles. The workspace is limited by surrounding walls, which are the only objects that could be considered as obstacles in classical motion planning. However, we do not assume a fixed size, as long as the workspace stays finite and rectangular.

For planning we work in the configuration space of the 2-dimensional special Euclidean group $SE(2) = \mathbb{R}^2 \times \mathbb{S}^1$. When only considering one cube, the group consists of the position in \mathbb{R}^2 and an orientation $\mathbb{S} = [0, 2\pi)$ [6]. When working with n cubes, the dimension of our configuration space increases to $\mathbb{R}^{2n} \times \mathbb{S}^1$. Note that we can still assume only one orientation for n cubes, because we are working with a global magnetic field orienting all cubes the same way. Figure 2.2 shows a configuration with four cubes in the workspace. It is irrelevant which exact physical cube is at which position as long as they are the same type, so switching the positions of the two red cubes in Figure 2.2 would lead to the same configuration as before.



Figure 2.3: Examples of polyominoes and their equality. a) and d) are equal, only the magnetic field changed its orientation. a) and c) are not equal, they have the same shape but rotated. a) and c) are also not equal because of different cube types in the same shape. e) shows an invalid polyomino in its grid representation (top) and how it behaves in the simulation (bottom).

2.3 Polyominoes

The embedded magnets not only align the cube with the magnetic field, they also allow cubes to self-assemble into polyominoes. Two cube faces can connect if their magnets have opposite polarities. Because of this and the alignment with the magnetic field, cubes can either be connected at north and south faces, or east and west faces, if the cubes are not the same type. A polyomino is a set of uniformly sized cubes on a 2-dimensional grid. Because we work with arbitrary positions and orientations the grid alignment does not hold true for multiple polyominoes in the workspace, but for each polyomino on its own the cubes can be represented in a local coordinate system with position (x, y) , $x, y \in \mathbb{Z}$ [9].

We consider fixed polyominoes, meaning that two polyominoes are distinct if their shape or orientation are different [9]. The magnetic field always provides an orientation, so in Figure 2.3 a) and d) the polyominoes are equal, just the magnetic field is rotated. Conversely, the polyominoes in Figure 2.3 a) and c) are the same shape but with a different rotation under the same magnetic field orientation, so they are not equal. Furthermore, two polyominoes are only equal if all the cubes at equal positions are the same type. The polyominoes in Figure 2.3 a) and b) are not equal because the cube types differ.

The size of a polyomino is the number of cubes it consists of. Because it is easier to view all structures in the workspace as a polyomino, single cubes are often referred to as trivial polyominoes with size 1. Although it is not possible to connect cubes of same type at east and west faces, the magnetic modular cubes can assemble structures like the one shown in Figure 2.3 e). The connection of the bottom two cubes is strong

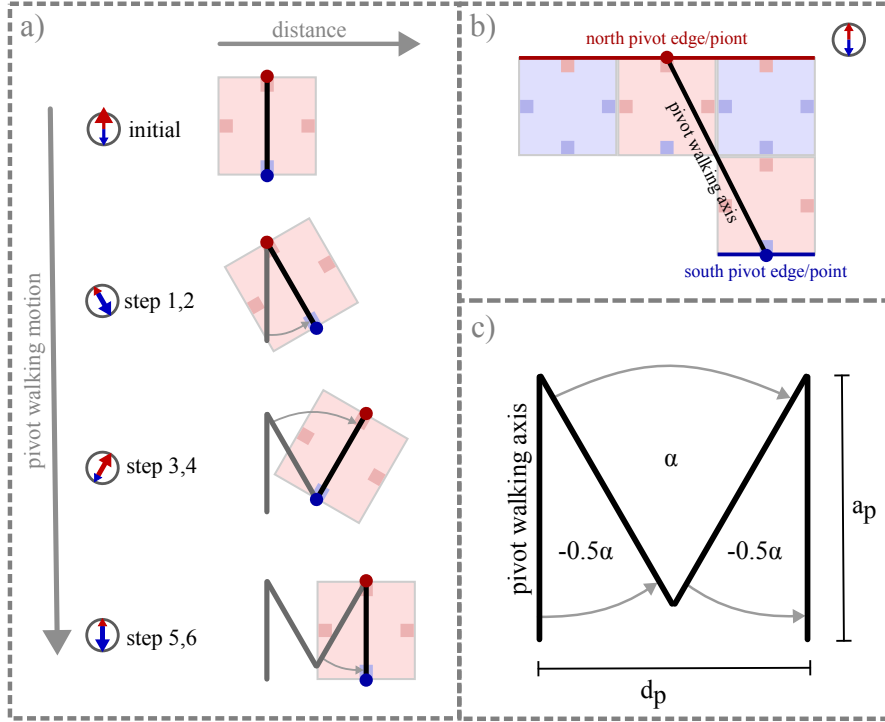


Figure 2.4: This Figure describes the pivot walking motion in detail. a) shows the six pivot walking steps for a single red cube. You can see the orientation of the magnetic field (bigger arrow indicates elevation). In b) an example polyomino with its pivot axis, edges and points is shown. c) illustrates the rotation of the pivot axis labeled with all the pivot walking parameters.

enough to hold the structure together, even though the four blue cubes on the top repel each other. The resulting polyomino in its grid representation has two east-west connections between cubes the same type and is therefore marked as an invalid polyomino. Because construction of invalid polyominoes is hard to handle, we omit any configuration containing them in our planning algorithms.

2.4 Motion Modes

In [4] three motion modes are presented. Rotation, pivot walking and rolling.

If the magnetic field orientation lays in the plane of the workspace and rotates without any inclination the rotation is performed around the center of mass for all polyominoes and we consider this motion a normal rotation.

Rotating the magnetic field perpendicular to the workspace plane, cubes can roll forwards or backwards. This rolling motion becomes problematic for self-assembly, because the top and bottom face of the cube, which contain no magnets, can become a side face.

2 Preliminaries

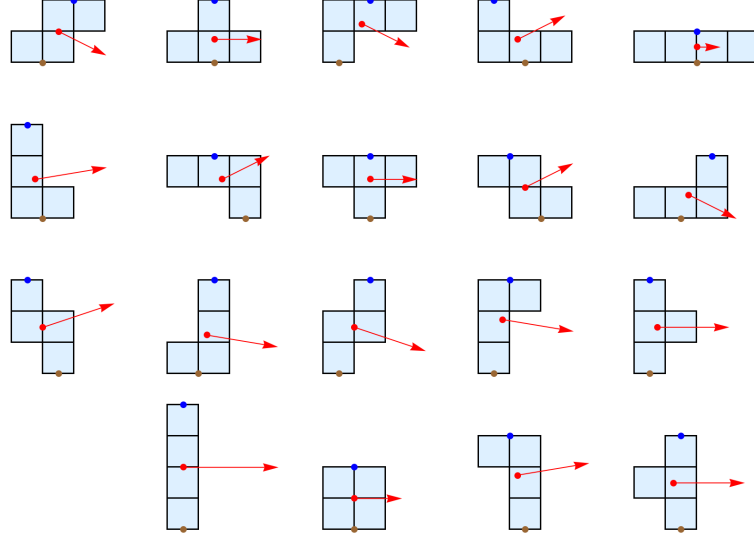


Figure 2.5: All 19 four-cube polyomino shapes with their displacement vector \vec{d} for one pivot walking cycle with $\alpha = \frac{\pi}{4}$. \vec{d} is drawn from the center of mass (red dot). North and south pivot point are drawn as blue and brown dots.

Because rotation and pivot walking are sufficient to reach any position in the workspace, we do not consider rolling in our simulation and planning algorithms.

When elevating the magnetic field orientation by lifting up the south pole slightly, all polyominoes will pivot on the north face bottom edges of their most north-placed cubes. Pulling up the north pole does the opposite. The polyominoes will pivot on the south face bottom edges of their most south-placed cubes. The sum of all these cube edges is called the north or south pivot-edge and by keeping the magnetic field elevated and rotating around the normal vector of the workspace plane, the polyominoes will rotate around the center point of their pivot-edge. This point is called the north or south pivot-point. All these edges and points are illustrated in Figure 2.4 b).

pivot walking: Not rotating around the center of mass is important for pivot walking. In the first step of a pivot walking cycle, the magnetic field is elevated to let the polyomino pivot on its north pivot edge. As a second step a rotation of $-\frac{1}{2} \cdot \alpha$ is performed around the north pivot point. $-\pi \leq \alpha \leq \pi$ is the pivot walking angle. For step 3 and 4 the elevation changes to its opposite to perform a rotation of α around the south pivot point. Step 5 and 6 are equal to 1 and 2 and will bring the polyomino back to its original orientation. You can see the pivot walking cycle steps in Figure 2.4 a) and have a closer look at its parameters in Figure 2.4 c).

After one pivot walking cycle, the polyomino has moved by a displacement vector \vec{d} with $\|\vec{d}\| = d_p$, so d_p is the distance the polyomino moved. The direction and length of \vec{d} changes with the shape of the polyomino. The movement is always perpendicular to

the pivot walking axis \vec{a} with $\|\vec{a}\| = a_p$, which is the vector between the north and the south pivot point, visualized in Figure 2.4 b). d_p can be calculated as

$$d_p = \sin\left(\frac{1}{2} \cdot \alpha\right) \cdot a_p.$$

To calculate \vec{d} you can take the perpendicular of \vec{a} and scale it to the length d_p .

To do a pivot walk left or right, you can choose a negative or positive α . When you choose a big α according to amount, d_p becomes also bigger, but the polyomino needs more space to the north and south to perform the rotations. For better maneuvering smaller values of α are preferable. There is a strong deviation of length and direction of the displacement for different polyomino shapes. So moving a pivot walking step might not move two polyominoes in the exact same direction. Figure 2.5 shows all 19 four-cube polyomino shapes with their displacement vectors.

3 Local Planner

This chapter is about the local planner that will be used for motion planning on a global scale in chapter 4. Local planning means that the planner only focuses on one concrete task. The task could be, to develop a plan that moves a polyomino from position a to position b , or even simpler, to develop a plan for one pivot walk. Since on a global scale we consider the problem of self-assembly, we are not really interested in changing positions alone. To work efficiently with local plans in the global planner, the initial and goal configuration of those plans should differ in the set of polyominoes they contain. The local planner takes two cubes out of different polyominoes and a valid edge-pair as inputs and attempts to establish a connection. If a local plan was successful, this guaranties a change of polyominoes in the workspace.

For this the local planner makes use of our simulator from chapter 5 in a closed-loop manner, meaning that the state of the simulation can be observed at any time and the actions can be adjusted accordingly. The local planner observes the position of cubes and polyominoes and works with the distance between them. It also works with the orientation of cube faces. In a real application of Magnetic Modular Cubes a camera, able to track cubes in the workspace, could be used to retrieve the necessary information.

The following sections 3.1 to 3.4 explain the techniques used in the local planning algorithm of section 3.5.

3.1 Connecting Polyominoes

The two main rules that have to be considered when connecting two polyominoes are: First, cubes can only be connected in the way described in section 2.3, so we can distinguish between east-west and north-south connections. Second, pivot walking (section 2.4) only allows the polyominoes to move left, in the direction of their west faces, or right, in the direction of their east faces. To move in any arbitrary direction it is necessary to rotate the polyomino, so that the east or west faces point into the desired direction. Keep in mind that a pivot walking cycle is not always performed in the exact east or west, depending on the polyomino shape.

East-west connections are generally easier to handle. If we want to connect an east face of polyomino A to a west face of polyomino B , A has to walk into the east direction towards B , or the other way around. When A should be connected at a south face of B , A can now walk into east or west direction towards B , or B could again do the opposite. A closer look on why these movement directions are important and how all of them can be established at any position is taken in section 3.2.

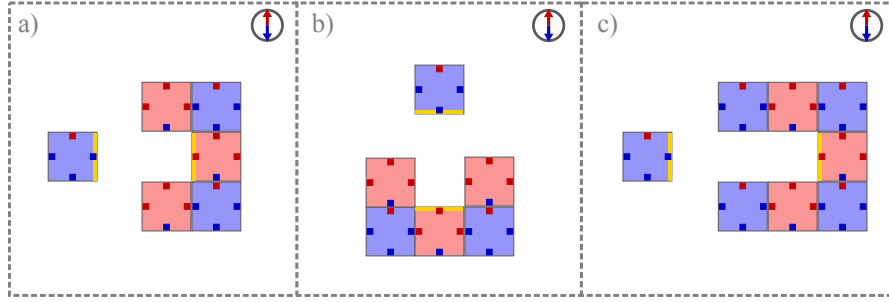


Figure 3.1: Three different examples for connecting polyominoes into holes. a) and b) show one-cube-deep holes (a) east-west and b) north-south). c) illustrates a two-cube-deep east-west hole. The edges to be connected are marked yellow.

Connecting In Holes: Connecting two polyominoes where one of the connection-faces is located inside a hole, is a difficult task in the continuous world. We differentiate between east-west and north-south located holes. Furthermore a hole can be of a certain depth measured in multiples of $2r_C$. Figure 3.1 shows examples for different holes with different depths.

Connecting into a hole with a depth of more than one times $2r_C$ is not possible. For instance, when inserting the blue single cube into the polyomino in Figure 3.1 c), the blue cube would connect with north and south faces of the polyomino before even reaching the full depth of the hole. But even holes with depth $2r_C$ are hard to handle. Inserting into a hole can be done by pivot walking, this would only work for east-west holes, or by letting magnetic forces attract the connection-faces. Relying on magnetic forces alone seem promising, since it would work for both hole types, but in reality not only the forces of the connection-faces are present. All the forces between other magnets prevent an easy slide-in. In our simulator the connection-face will be more attracted or repelled by faces outside the hole, then the once inside. Pivot walking into east-west holes, even with small values for α , has also a high failure rate because of other magnets. We generally try to avoid local plans that are connecting polyominoes inside holes.

3.2 Aligning Cubes

3.3 Walking And Waiting

3.4 Plan And Plan-State

3.5 Local Planning Algorithm

4 Global Planner

5 Simulator

Bibliography

- [1] A. Becker, E. D. Demaine, S. P. Fekete, G. Habibi, and J. McLurkin. Reconfiguring massive particle swarms with limited, global control. In P. Flocchini, J. Gao, E. Kranakis, and F. Meyer auf der Heide, editors, *Algorithms for Sensor Systems*, pages 51–66, Berlin, Heidelberg, 2014. Springer Berlin Heidelberg.
- [2] A. Becker, E. D. Demaine, S. P. Fekete, and J. McLurkin. Particle computation: Designing worlds to control robot swarms with only global signals. In *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6751–6756, 2014.
- [3] 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. *Algorithmica*, 82(2):165–187, Feb 2020.
- [4] A. Bhattacharjee, Y. Lu, A. T. Becker, and M. Kim. Magnetically controlled modular cubes with reconfigurable self-assembly and disassembly. *IEEE Transactions on Robotics*, 38(3):1793–1805, 2022.
- [5] J. Bishop, S. Burden, E. Klavins, R. Kreisberg, W. Malone, N. Napp, and T. Nguyen. Programmable parts: A demonstration of the grammatical approach to self-organization. In *2005 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3684–3691. IEEE, 2005.
- [6] S. M. LaValle. *Planning Algorithms*. Cambridge University Press, Cambridge, U.K., 2006. Available at <http://planning.cs.uiuc.edu/>.
- [7] S. M. LaValle et al. Rapidly-exploring random trees: A new tool for path planning. 1998.
- [8] S. M. LaValle and J. J. Kuffner. Rapidly-exploring random trees: Progress and prospects: Steven m. lavallo, iowa state university, a james j. kuffner, jr., university of tokyo, tokyo, japan. *Algorithmic and computational robotics*, pages 303–307, 2001.
- [9] Y. Lu, A. Bhattacharjee, D. Biediger, M. Kim, and A. T. Becker. Enumeration of polyominoes and polycubes composed of magnetic cubes. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 6977–6982, 2021.

Bibliography

- [10] A. Mueller. Modern robotics: Mechanics, planning, and control [bookshelf]. *IEEE Control Systems Magazine*, 39(6):100–102, 2019.
- [11] A. T. B. Patrick Blumenberg, Arne Schmidt. Computing motion plans for assembling particles with global control. In *Under Review*. IEEE, 2023.
- [12] R. Pelrine, A. Wong-Foy, A. Hsu, and B. McCoy. Self-assembly of milli-scale robotic manipulators: A path to highly adaptive, robust automation systems. In *2016 International Conference on Manipulation, Automation and Robotics at Small Scales (MARSS)*, pages 1–6. IEEE, 2016.
- [13] W. Saab, P. Racioppo, and P. Ben-Tzvi. A review of coupling mechanism designs for modular reconfigurable robots. *Robotica*, 37(2):378–403, 2019.
- [14] 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*, 3(4):3521–3528, 2018.
- [15] 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*, 103(2):205–224, 2015.
- [16] P. J. White and M. Yim. Scalable modular self-reconfigurable robots using external actuation. In *2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2773–2778. IEEE, 2007.
- [17] E. Winfree. *Algorithmic self-assembly of DNA*. California Institute of Technology, 1998.