Technology
Arts Sciences
TH Köln

On 3D fixed-angle chains that are locked, equilateral, equiangular, and obtuse

Computer Engineering

Faculty of Information, Media and Electrical Engineering

Technical University of Cologne

Master Thesis

for the award of the academic degree

Master of Science

submitted by

Jonas Lummerzheim

First Examiner: Prof. Dr. Hubert Randerath

Second Examiner: Prof. Dr. Dieter Rosenthal

Year of Publication: 2022

Declaration of Lieu of Oath

I certify that I have written the submitted work independently. All passages taken verbatim or in spirit from published or unpublished works of others or of the author himself/herself have been marked as taken. All sources and aids that I have used for the work are indicated. The work has not been submitted to any other examination authority with the same content or in essential parts.

Place, Date, Signature _____

Master Thesis

Title: On 3D fixed-angle chains that are locked, equilateral, equiangular, and obtuse

Examiner:

- Prof. Dr. Hubert Randerath
- Prof. Dr. Dieter Rosenthal

Abstract:

For most classes of chains, it is known if these contain locks, but especially for fixed-angle equilateral equiangular obtuse open polygonal chains in 3D, which can be used to model protein backbones, this is unknown. [1] [2] Fixed-angle equilateral equiangular obtuse closed and open polygonal chains can be used to model polymers. [3] For these, it is clear, that locks based on knots exist, but not which chains are generally locked. We therefore examine both open and closed fixed-angle equilateral equiangular obtuse chains. For this purpose, those chains are divided into various subgroups and, depending on the subgroup, other aspects are investigated to show locks. Techniques from knot theory, graph theory, and specifically robot arm reachability and motion planning are combined. Algorithms are developed to create chains in desired configurations and to study them. It is shown why all fixed-angle equilateral equiangular obtuse closed chains are expected to be locked or in rare cases rigid and non-locked, but never non-locked and non-rigid. For fixed-angle equilateral equiangular obtuse open chains it is shown why it is expected that there are open chains that are locked and that the smallest locked open chain has $n = 7$.

Keywords: 3D, Fixed-Angle, Polygonal Chain, Locked, Equilateral, Equiangular, Obtuse

Table of Contents

1. Introduction 1
1.1. Motivation 1
1.2. Scope of Application 1
1.3. Structure and Objectives 2
1.3.1. Questions of Interest 2
2. Theoretical Foundation 3
2.1. Fundamentals 3
2.2. Knots 8
2.3. Chains with Universal Joints 9
2.4. Fixed-Angle Linkages 11
2.5. Equilateral Fixed-Angle Chains 11
2.6. Robot Arm Reachability, Motion Planning and Configuration Space 12
3. Unity 15
4. Impact of Radius 16
5. Closed Chains 17
5.1. Regular Polygons 17
5.2. Locked Closed Chains 17
5.3. Closed Chains Summary 26
6. Open Chains 27
6.1. At First Glance 27
6.2. Rigidity 28
6.3. Unfolding Resistance 30
6.4. Simple Locks 41
6.5. Configuration Space 44
6.6. Producible Protein Chains 50
6.7. Cutting Chains 51
6.8. Interlocked Open Chains 52
6.9. Open Chains Summary 52
7. Conclusion 54
8. Bibliography 55

List of Tables

Table 1: Summary of results. The `___' means no restriction of the type indicated in the column heading. Entries marked `?' are open problems 11

Table 2: Observations for physics models based on simple locks with l=2 and r=0.025 43

List of Figures

Figure 1: α-angle 4

Figure 2: φ-angle 4

Figure 3: Reidemeister moves and coloring of unknot and trefoil knot 9

Figure 4: Three separated configuration-subspaces, unknotted, and two separated prime knots, 3_1, also called trefoil knot, and 6_2. 18

Figure 5: Trefoil and unknotted closed models with m=48, l=2, r=0.1, and α=135°. 18

Figure 6: Smallest knot. 19

Figure 7: Three physics models generated using Algorithm 1 with l=2, r=0.025, m=9, and α=95°. The two left models started with an all-zero and the right with zig-zag configuration. 21

Figure 8: Two physics models generated using Algorithm 1 with l=2, r=0.025, m=13, the left has α=125° and the right α=124° 22

Figure 9: Three physics models generated using Algorithm 1 with l=2, r=0.025, and α=95°. The left model with n=6, the middle with n=7, and the right with n=8. 22

Figure 10: First half of eight distinctive configurations with equal spacing on the circle of the configuration subspace of a wooden model with m=8, l=19, r=0.15, and α≈95°. 23

Figure 11: Two physics models in separated configuration cycles with l=2, r=0.025, m=7, and α=95°. 24

Figure 12: Rigid closed physics model with l=2, r=0.025, m=6, and α=95°. 25

Figure 13: First locked open chain idea as wooden model with m=22, l=19.5, r=0.15 and α≈98° 27

Figure 14: Model based on algorithm 2 with m=30, l=2, r=0.1, α=120°, β=60° and γ=3° 28

Figure 15: Model with m=29, l=2.04, r=0.1, α=95° and markers and a wooden copy with n=29, l=19.5, r=0.15, α≈95°, and distance holders. Start node is blue, end node is pink, left constraint is teal, and right constraint is green. 29

Figure 16: Left m_1 with m=514, α=93°, and right m_2 with m=613, α=113.1°, both with l=2.04 and r=0.1 31

Figure 17: number of edges in distance one or less per distance from center for each edge of m_1 with l=2.04, r=0.1, m=514, and α=93° 31

Figure 18: number of edges in distance one or less per distance from center for each edge of m_2 with l=2.04, r=0.1, m=613, and α=113.1° 32

Figure 19: Distance to center per edge index for m_1 with l=2.04, r=0.1, m=514, and α=93° 32

Figure 20: Distance to center per edge index for m_2 with l=2.04, r=0.1, m=613, and α=113.1° 33

Figure 21: Frequency of distances from center point for edges of m_1 with l=2.04, r=0.1, m=514, and α=93° 33

Figure 22: Frequency of distances from center point for edges of m_2 with l=2.04, r=0.1, m=613, and α=113.1° 34

Figure 23: Constraints per distance from center in range 0.01 for m_1 with l=2.04, r=0.1, m=514, and α=93° 34

Figure 24: Constraints per distance from center in range 0.02 for m_1 with l=2.04, r=0.1, m=514, and α=93° 35

Figure 25: Constraints per distance from center in range 0.05 for m_1 with l=2.04, r=0.1, m=514, and α=93° 35

Figure 26: Constraints per distance from center in range 0.1 for m_1 with l=2.04, r=0.1, m=514, and α=93° 35

Figure 27: Constraints per distance from center in range 0.01 for m_2 with l=2.04, r=0.1, m=613, and α=123.1° 35

Figure 28: Constraints per distance from center in range 0.02 for m_2 with l=2.04, r=0.1, m=613, and α=123.1° 36

Figure 29: Constraints per distance from center in range 0.05 for m_2 with l=2.04, r=0.1, m=613, and α=123.1° 36

Figure 30: Constraints per distance from center in range 0.1 for m_2 with l=2.04, r=0.1, m=613, and α=123.1° 36

Figure 31: m_3 with m=24, l=2, r=0.025, α=91°, start-β=30°, and start-γ=3° 38

Figure 32: number of edges in distance one or less per distance from center for each edge of m_3 with l=2, r=0.025, m=24, α=91°, start-β=30°, and start-γ=3° 39

Figure 33: Distance to center per edge index for m_3 with l=2, r=0.025, m=24, α=91°, start-β=30°, and start-γ=3° 39

Figure 34: Frequency of distances from center point for edges of m_3 with l=2, r=0.025, m=24, α=91°, start-β=30°, and start-γ=3° 39

Figure 35: Constraints per distance from center in range 0.01 for m_3 with l=2, r=0.025, m=24, α=91°, start-β=30°, and start-γ=3° 40

Figure 36: Constraints per distance from center in range 0.02 for m_3 with l=2, r=0.025, m=24, α=91°, start-β=30°, and start-γ=3° 40

Figure 37: Constraints per distance from center in range 0.05 for m_3 with l=2, r=0.025, m=24, α=91°, start-β=30°, and start-γ=3° 40

Figure 38: Constraints per distance from center in range 0.1 for m_3 with l=2, r=0.025, m=24, α=91°, start-β=30°, and start-γ=3° 40

Figure 39: Knitting needles 41

Figure 40: Wooden model with m=6, l=19, r=0.15 and α≈92°. 42

Figure 41: Smallest locked open physics model with l=2, r=0.025, m=6, α=94° and $\Phi_{\text{start}} = (-45°, 45°, -34°, 12°)$ 42

Figure 42: Crossing α-range for first and last edge based on the regular polygon equation and locked α-range rounded up with 0.5° accuracy at the borders for physics models based on knitting needles with l=2 and r=0.025 43

Figure 43: Open model with m=6, α=94°, l=2, r=0.05 and Φ = (-45°, 45°, -34.2°, 17°). 46

Figure 44: Three perspectives of the 4° hypercube overlay in 3D using (j1, j2, j3) as coordinates with $stepsize$=0.1° of a chain with m=6, α=94°, l=2, r=0.05 and Φ_{start}= (-45°, 45°, -34.2°, 17°), where boarders of the local configuration subspace are shown. 46

Figure 45: Open model $m4$ with m=6, α=97°, l=2, r=0.08 and Φ = (-22°, -2°, 20°, -45°). 48

Figure 46: Slices j4=-45°, j4=-47.5° and j4=-50° of the configuration space results of $m4$..... 48

Figure 47: Overlay of all j4 slices of the configuration space results of $m4$..... 49

Figure 48: Overlay of all j1, j2, j3, and j4 slices where only coordinates without self-intersection are displayed of the configuration space results of $m4$. 49

1. Introduction

1.1. Motivation

We are currently living in a difficult time in which new and reliable vaccines and medicaments must be developed at high speed. To be able to save the ecosystem earth, alternatives must be found quickly in many areas like food and material production. Proteins and polymers are playing an increasingly important role in this. [4] To increase the speed of finding new proteins and polymers, we are investigating 3D fixed-angle chains that are locked, equilateral, equiangular, and obtuse.

Knowledge gained in this field is applicable in many ways and overlaps with my professional as well as private interests. Examples are robotics, one-armed mazes, knots or weavings, origami, and Langton's ant.

1.2. Scope of Application

Closed and open 3D fixed-angle chains that are equilateral, equiangular, and obtuse can be used to model the geometry of polymers [3] and the open of these chains can be used to model the geometry of protein backbones, where chemical bonds are based on obtuse angles that can be modeled roughly with equiangular and equilateral restrictions [5]. It is assumed that ribosomes are only able to produce protein backbones that are flattenable. [5] For open chains, it would be helpful to know whether all chains are flattenable or if not, which are not, because then the search space for foldable protein backbones can be reduced. [6]

Macromolecular structures have a significant effect on transfection[1] efficacy. [7] Cyclic knot polymer structures can have higher transfection capabilities and lower toxicity then cyclic unknotted polymer structures. [7] [8]

These can be produced in a knot configuration in vitro[2] and then unfolded in vivo[3]. Tests in the adult rat brain demonstrated the usage for neurodegenerative disease therapies.

[9]

Even though robot arms in most cases do not have fixed-angle, equilateral, equiangular, and obtuse restrictions, there search spaces for motion planning can be reduced in special cases using our results.

---

[1] *"(In genetic engineering) Any process by which eukaryotic cells take up foreign DNA in a form that is replicated and passed to daughter cells, thereby changing the genetic constitution of the cell line or organism concerned. It is synonymous with transformation in bacteria. Various techniques have been devised, depending on the nature of the recipient cell. The most popular are the use of liposomes, electroporation, microinjection, and biolistics."* [10] (Oxford Reference)

[2] *"Designating biological processes made to occur experimentally in isolation from the whole organism; literally "in glass," i.e., in the test tube. Examples: tissue cultures, enzyme-substrate reactions. Contrast with in vivo, ex vivo."* [11] (Oxford Reference)

[3] *"Literally, in life, but used generally for procedures or tests done on intact organisms rather than on isolated cells in culture (in vitro)."* [12] (Oxford Reference)

The minimum number of edges of a closed chain, which are needed to form a knot, can possibly form another classification possibility for knots. Equilateral polygonal chains with universal joints are used by Kenneth C. Millett to model knots [3], but equiangular and fixed-angle restrictions are not given.

1.3. Structure and Objectives

We begin by looking at the existing literature on the topic to determine which gaps need to be filled and what is already known. We are not only looking at the corresponding chains, but also at related types of chains, knots that could be in a chain, algorithms for robotic arms that could be helpful in the investigation, and the connection of the chains to bio-chemical processes.

3D fixed-angle closed chains that are locked, equilateral, equiangular, and obtuse should be classified. For this we will split these chains into smaller groups and try to show for them why these are locked or not locked. We summarize the results to solve the problem.

The most important goal of this work is to answer whether 3D fixed-angle open chains that are locked, equilateral, equiangular, and obtuse exist. Because it is not clear whether this is the case, various ideas are being tried out. If a lock is found, we will classify it, investigate whether the results are relevant to the scope of application, and investigate their applicability.

Found results are either proven or it is explained what needs to be proven. In the process, a roadmap is created. Solving this roadmap completes the core of the topic.

In addition, there is a small insight into the peripheral areas of cutting chains and interlocked chains.

1.3.1. Questions of Interest

The following seven questions provide a rough structure that we work through. Questions 2, 3 and 4 help us to answer question 1.

Question 1: Which closed chains are locked? classify locked closed chains.

Question 2: Which chains can be closed?

Question 3: Which closed chains can be used to tie a non-trivial knot?

Question 4: Which closed chains can have a self-intersection?

Question 5: Does a locked open chain exist? If yes, classify locked open chains. This question is asked by Erik Demaine. [1] [2]

Question 6: Are there classes of open chains that cannot be produced by a ribosome? Answered by Erik Demaine, however, here again considered for found locked chains. [13]

Question 7: How many cuts are necessary to unlock a chain or all resulting chains? This question is an adaption of a question from Anna Lubiw. [14]

2. Theoretical Foundation

Chains studied here belong to an independent superclass of polygonal chains with fixed-angle restriction under dihedral motion in 3D. Basically, this means that results for polygonal chains with universal joints are in most cases not directly applicable to the chains studied here. All other restrictions used describe subclasses of polygonal chains with fixed-angle restriction under dihedral motion in 3D. This chapter looks at the existing literature which is useful in the context of this work and does not give a summary about all classes of polygonal chains. For a summary on most common classes of locked polygonal chains, the book "Geometric Folding Algorithms: Linkages, Origami, Polyhedra" [15] is recommended.

2.1. Fundamentals

This chapter defines the vocabulary used.

Linkage

A linkage consists of a graph $G = (V, E)$ and edge lengths $l : E \to \mathbb{R}_{\geq 0}$. G is called the structure graph of the linkage. [16]

Chain

An open (polygonal) chain of length m is a linkage whose structure graph is $G = (V, E)$ where $V = \{v_1, v_2, \dots, v_{m+1}\}$ and $E = \{(v_1, v_2); (v_2, v_3); \dots; (v_m, v_{m+1})\}$.

A closed (polygonal) chain of length m is a linkage whose structure graph is $G = (V, E)$ where $V = \{v_1, v_2, \dots, v_m\}$ and $E = \{(v_1, v_2); (v_2, v_3); \dots; (v_{m-1}, v_n); (v_m, v_1)\}$.

A chain is either an open or a closed chain.

Because isolated fixed-angle chains in Euclidean 3D that are non-self-intersecting, equilateral, equiangular, and obtuse are considered here, these properties are implicitly associated with the term chain in all following chapters (except subchapters of 2. Theoretical Foundation).

If m is used anywhere the number of edges is associated with it. n corresponds to the number of vertices, which is not used is most cases.

Model

In some Situations, it is useful to have a radius on the edges. If a radius function is defined it is equiradial. A chain with radius on the edges will be called model. There are five different types of models:

- Model: Virtual model which cannot be used for physics simulations. This model lives in an environment without gravity and is not moving. Therefore, time-consuming algorithms can be used to place edges in a sensible way, which are not applicable during a physics simulation.
- Physics Model: Virtual model with rigid body and joints. This model lives in an environment with gravity and collision detection.

- Wooden Model: Real world model crafted with lumber. Not really accurate, but it offers the possibility to hold small models in your hand, to simply test what happens when a φ-angle is changed or to visualize an idea.
- 3D printed Model: Like a wooden Model, but more accurate.
- Rope Model: Real world model based on a rope. Just used to visualize knots.

A model can be translated into a physics model and vice versa.

α-Angle

α is an angle function that specifies the smallest angle between any two adjacent edges. Because α must be obtuse $\frac{\pi}{2} < \alpha < \pi$ applies. Figure 1 shows an example.

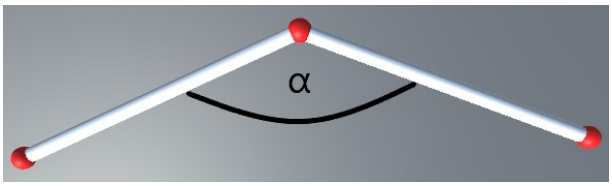

Figure 1: α-angle

φ-Angle

φ is an angle function that specifies the angle between two planes $\Pi_1\ (e_1, e_2)$ and $\Pi_2\ (e_2, e_3)$ which are spanned by any three consecutive edges e_1, e_2 and e_3. φ has no influence on α. It applies that $-\pi < \varphi \leq \pi$. A motion that changes only φ-angles is called dihedral motion. [17] A dihedral motion is called local if, all joints, except one, are fixed. [18] Figure 2 shows an example. For $\varphi = 0$ the edges e_1, e_2 and e_3 are on the same plane. If it is assumed that e_2 lies in this plane on the x-axis, then e_1 and e_3 proceed both in the positive or both in the negative y-direction in that case. For simplification, only the edge on which the joint is placed, or the joint index must be passed to the φ-function. The set of all dihedral angle definitions is called Φ. For closed chains Φ has m elements and for open chains Φ has $m - 2$ elements.

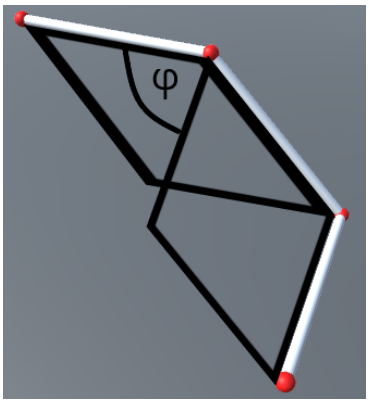

Figure 2: φ-angle

Universal Joint

Universal joints do not have a fixed-angle restriction.

Limited Joint

A dihedral joint is limited, if that joint is not allowed to use its whole configuration space. A non-limited joint has a range of $-\pi < \varphi \leq \pi$. If the range is smaller this joint is limited.

Isolated

If in an infinite 3D space, there is only one chain or model and no obstacles except the chain or model itself this chain is called isolated. Inside a physics simulation this restriction is partially violated, and a plate is added as an obstacle to perform tests.

Fixed-Angle

All α-angles are fixed. Only φ-angles may change.

Non-Self-Intersection

All edges of a linkage do not cross.

Equilateral

All lengths $l(e) \mid \forall e \in E$ are equal. Because l now no longer has any influence on the structure, $l(e) = 1 \mid \forall e \in E$ can be used. [16] l will be defined anyway to specify the ratio between length, radius, and other distances. Unit lengths on edges correspond to equilateral. All length, radius, and distance specifications for chains are unitless.

Equiangular

All α-angles are equal.

Obtuse

If $\frac{\pi}{2} < \alpha < \pi$ is true, α is obtuse.

Equiradial

The radii of all edges are equal.

Configuration

A chain is defined by its α-angle and number of edges. The configuration of a chain is determined by its φ-angles. If all φ-angles of two equally defined chains are also equal, only then both chains are in the same configuration.

Configuration Space

The configuration space includes all valid configurations of a defined chain. Only those configurations do not belong to the configuration space, which lead to self-intersections.

Configuration Subspace

If the configuration space of a chain is not completely connected, it consists of several components or configuration subspaces.

Lock

If a chain has a separated configuration space, it is called locked. In detail this means, if there exists a configuration that cannot be transformed to every other possible valid configuration of the same chain without violating its tree structure or other restrictions, this chain is called locked. Any open chain has a flattenable configuration. [19] If a non-flattenable configuration of an open chain is found, a lock is clearly proven. The same applies to closed chains if they have a valid flat-state. An alternative proof for closed chains is to show a configuration with a simple projection and a knot. Lock is equated with non-flattenable by Erik Demaine et. al. [13] We define lock as a configuration space that consists of multiple disconnected configuration subspaces which is based on another definition of locks given by Erik Demaine et. al. [20]

Flat-State

If a chain configuration is planar, this is called flat-state of a chain. The same applies to a model that forms a flat-state chain when the radius is removed.

Flattenable

If the current configuration of a chain or model is in the same configuration-subspace as any flat-state, this is called flattenable. If the current configuration of a chain or model is in a different configuration-subspace, this is called non-flattenable.

Flat-State Connected

If all flat-states of a chain are in the same configuration subspace this is called flat-state connected. If this is not the case this is called flat-state disconnected.

Rigid

If the configuration of a chain is in a configuration subspace that has only one element this chain is called rigid.

Knot

In mathematics a knot is an embedding of a circle into 3D Euclidean space. [21]

Simple Projection

A chain has a simple projection if the projection of a 3D chain onto a 2D surface has no intersections. Non-trivial knots never have a simple projection. [22]

Radius and Length

$r : E \to \mathbb{R}_{\geq 0}$ is a function that specifies the radius for edges and $l : E \to \mathbb{R}_{\geq 0}$ is a function that specifies the length for edges. If a radius is defined, a length must also be defined to give the relation between both values.

ε-Error

ε is used for very small errors or differences.

Cut

A process to create an open chain from a closed chain or to create two open chains from an open chain is called a cut. A vertex is added, and this vertex replaces a vertex of an existing edge.

Constraint

Unconstraint means, that there are no other edges near the examined edge. Right constraint means, that there is at least one other edge on the right side of the examined edge and on its left side are no edges. Left constraint means same as right constraint, just for the left side. Full constraint is given when there are edges at the right and left side. The plain $\Pi(e_1, e_2)$ formed by two consecutive edges e_1 and e_2 gives the boundaries of left and right for e_2. The boundary of the first edge is formed with the help of an imaginary edge placed with first $\varphi(e_1) = 0°$. When you look in the direction of e_2 on the axis of e_1 and e_2 leads downwards, then right of e_2 is defined as your right side. If no range is defined, it means that edges are touching, else that there is at least one edge touching or in the distance defined by a given range.

Pseudo Code

Content of algorithms is given in pseudo code based on python indentation and syntax.

2.2. Knots

In mathematics a knot is an embedding of a circle into 3D Euclidean space and therefore knots are always closed. Two knots are defined to be equivalent if there is an ambient isotopy between them, or in simple words, if one knot can be deformed to the other knot without cutting it. There is usually no ambient isotopy between a knot and its reflection in 3D. A central tool for the examination of knots is a diagram, that is a regular projection of a knot onto a 2D surface, where the intersection points are marked in the projection in such a way that it is possible to see which section runs above another. The lower section is displayed separated and the upper one is connected.

[21] [23]

A regular projection is given, if the resulting image has only crossings, where two sections meet. [24]

There is a simple connection between ambient isotopic and configuration subspaces. If and only if two configurations are in the same configuration subspace, the knots created with these configurations are ambient isotopic.

To classify knots, they are divided into prime knots and composite knots, where composite knots are based on several prime knots that are cut and connected. Prime knots are described by two numbers, on the one hand the minimum number of intersections or crossings within all possible projections that a knot can have, and on the other hand an identification number within this group.

[25] [23]

Reflections of each other often carry the same identification numbers even if those are not ambient isotopic. A simple example for this is the trefoil knot, which, apart from its mirroring, is the only knot whose projection has at least three intersections, and therefore 3_1 is its designation.

[24] [21] [23]

Two diagrams of knots can be related by a finite sequence of Reidemeister moves, if and only if the two diagrams are ambient isotopic. There are three simple Reidemeister moves, which are applicable to both a diagram and a knot in 3D.

[24] [23] [26]

We call these twist, poke, and slide. [23] [26]

Coloring arcs in diagrams can be used to divide knots into two groups. The arcs of the diagram are colored in such a way that at all intersections either three arcs of the same color meet or three arcs of different colors meet. An arc is a section of the knot whose boundaries are where other sections pass over it in the diagram. Diagrams which can be colored with three colors are tricolorable and all others are not. A feature of all Reidemeister moves is that these do not change the coloring of a diagram. Based on the Reidemeister moves we know that all knots that are ambient isotopic to each other have the same coloring. The trefoil knot is tricolorable and the unknot is not.

[27]

There are different ways to classify knots, for example the Alexander-Conway polynomial [28], the Jones polynomial [29] [28], or the generalization of both, the HOMFLYPT polynomial [3], but for our purposes coloring and Reidemeister moves are sufficient. Figure 3 illustrates the Reidemeister moves and the coloring of the unknot and the trefoil knot. This proves that any equilateral equiangular fixed-angle closed chain is locked, if it can form an unknot and a trefoil knot. To show that a lock is present on an alternative way, two configurations of such a closed chain can be represented by rope models. If

these can be deformed in such a way that one is clearly an unknot and the other clearly a trefoil knot, a lock is present.

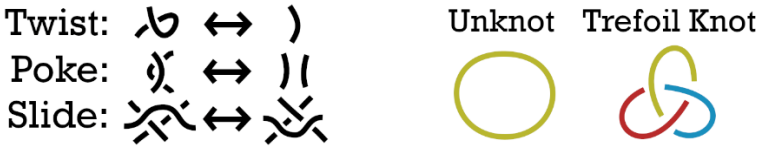

Figure 3: Reidemeister moves and coloring of unknot and trefoil knot

The Fary-Milnor Theorem tells us that the curvature of any smooth non-trivial knot is greater than 4π. [3] The trivial knot is the unknot. [27]

Knots are modeled by Kenneth C. Millet using equilateral closed polygonal chains with universal joints. Whether equilateral open polygonal chains with universal joints can be locked is unknown.

[30] [3]

To form the trefoil knot an equilateral polygonal chain with universal joints and at least six edges is necessary. Kenneth C. Millet defines total curvature of knots created with equilateral polygonal chains with universal joints as the sum of all exterior angles at the vertices.

[30]

Total curvature is equal for any two configurations of a fixed-angle equilateral equiangular obtuse closed chain based on any combination of n and α. We will use this in combination with the Fary-Milnor Theorem.

The probability that a random configuration of an equilateral closed chain with universal joints is a non-trivial knot goes to one exponential quickly based on n. [30]

The most important researchers of this topic are Kurt Reidemeister, who simplified the study of knots with his moves in 1920, James W. Alexander, who invented the Alexander polynomial in 1928, John Horton Conway, who invented the skein relation which is used in the Jones polynomial and the HOMFLYPT polynomial [3], Vaughan F. R. Jones, who invented the Jones polynomial in 1984 and eight people who discovered the HOMEFLYPT polynomial [3], where HOMEFLYPT [3] is derived from their initials. Building on their results, researchers have found connections from knot theory to graph theory, statistical mechanics, molecular biology, and quantum field theory.

[29]

The HOMFLYPT polynomial is often called HOMFLY Polynomial.

The calculation of HOMFLY polynomial is NP-Hard. [30]

2.3. Chains with Universal Joints

Even if the results of polygonal chains with universal joints are not directly applicable, much inspiration can be found there for the chains studied here.

Simple projections of open polygonal chains with universal joints are used by Biedl et. al. to detect if an open polygonal chain with universal joints can be straightened with efficient algorithms. A projection corresponds to the shadow of a chain cast on a flat surface. This is called simple only if the projection does not contain any self-intersections.

[31]

Simple projections may therefore be suitable for pre-probing whether an open chain is flattenable.

Knitting needles, shown in 6.4. Simple Locks, describes the smallest locked open polygonal chain with universal joints in 3D. Five edges relate to four universal joints to form a knot. The main point because knitting needles is locked is based on the length definitions of the edges and its structure graph.

[31]

Because we require that all edges are of the same length, all α-angles are equal and obtuse, knitting needles cannot be used. However, the search for the smallest “knot” is also useful for fixed-angle equilateral equiangular obtuse open chains.

Biedl et. al. show that locks also exist for trivial knots, based on double knitting needles, that is a closed chain with universal joints in the form of knitting needles, where this is formed from two knitting needles chains, of which start, and end nodes are connected.

[31]

This leads to the assumption that fixed-angle equilateral equiangular obtuse closed chains may be locked even if knots are excluded. Double knitting needles cannot be created with fixed-angle equilateral equiangular obtuse closed chains.

When chains with universal joints are convex, simpler algorithms can be applied in most cases. An open chain with universal joints is convex if a convex closed chain is formed by adding an imaginary edge. Algorithms to bring planar chains with universal joints into a convex shape are based on orthogonal planes, in which parts of chains within a plane are changed to prevent self-intersections.

[31]

Although these techniques are not directly applicable to chains with fixed-angle restrictions, convex features, or surfaces on which parts of a chain are placed can be useful tools for us as well.

Robert Connelly et. al. showed that if a chain with universal joints is rigid, it is locked. They proved it for chains with universal joints, based on the configuration space. In addition, chains that touch themselves are distinguished from chains that do not. These are called self-touching and non-self-touching.

[32]

An important point here is that the proof refers exclusively to the configuration space, so it is also clear that fixed-angle open chains are also locked if they are rigid. In addition, it seems useful to make a distinction between chains that can touch themselves or form self-intersections if allowed, and those that cannot.

Helmut Alt et. al. proved that the reconfiguration problem of polygonal chains with universal joints in 3D is PSPACE-hard. They compared it with the motion planning problem.

[33]

2.4. Fixed-Angle Linkages

For fixed-angle linkages in 3D different subclasses were examined by Aloupis et. al. to determine whether all flat-states of a linkage are located in the same configuration subspace. The results achieved so far are summarized in Table 1. From the results, it appears that all flat-states of fixed-angle nonacute open chains lie in the same configuration subspace. For fixed-angle nonacute closed chains this is an open problem.

[18]

This result can be directly applied to 3D fixed-angle open chains that are equilateral, equiangular, and obtuse. Even if there are no results for fixed-angle nonacute closed chains, for fixed-angle orthogonal equilateral closed chains all flat-states are in the same configuration subspace. [18] Orthogonal α-angles are not obtuse, but 135°-angles are obtuse and two consecutive angles in the same plane can result in an orthogonal turn. Therefore, it does not seem particularly promising to look for flat-states based on simple angles that are in different configuration subspaces to show locks for fixed-angle equilateral equiangular obtuse closed chains. To show locks the focus is therefore not set on flat-states.

For the proofs of these results, some dihedral joints were fixed and others not, and edges were placed on planes that are orthogonal to each other. [18] Both approaches can be helpful tools in the construction of chains or in proofs.

Table 1: Summary of results. The `___' means no restriction of the type indicated in the column heading. Entries marked `?' are open problems

Constraints on Fixed-Angle Linkage				Flat-state
Connectivity	Angles	Lengths	Motions	connectivity
Open chain	—	—	—	?
	has a monotone state		—	?
	nonacute	—	—	Connected
	equal acute	—	—	Connected [2]
	each in $(60°, 90°]$	unit	—	Connected [2]
	—	—	180° edge spins	Disconnected
	orthogonal	—	180° edge spins	Connected
Set of chains, each	orthogonal	—	—	Connected
pinned at one end	orthogonal	—	partially rigid	Disconnected
Closed chain	—	—	—	?
	nonacute	—	—	?
	orthogonal	—	—	?
	orthogonal	unit	—	Connected
Tree	—	—	—	?
	orthogonal	—	—	?
	orthogonal	—	partially rigid	Disconnected
Graph	orthogonal	—	—	Disconnected

Source: Table 1 in Flat-State Connectivity of Linkages under Dihedral Motions [18]

2.5. Equilateral Fixed-Angle Chains

Aloupis et. al. showed that all flat-states of fixed equilateral equiangular chains with angles in $0° < 2\alpha < 90°$ are connected by dihedral motions. [34]

Nadia Benbernou and Joseph O'Rourke have investigated the maximum span of fixed-angle chains. Span is the distance between the start and end nodes of an open chain. They have shown for fixed-angle equilateral chains that the configuration with the maximum span is the trans-configuration. A

chain is in the trans-configuration when its dihedral joints are alternately set to 0° and 180°. The trans-configuration is a flat-state and for fixed-angle equilateral chains in 3D this configuration can be found in O(1).

[19] [6]

On the one hand, this also defines the maximum span of fixed-angle equilateral equiangular obtuse chains; on the other hand, every of those open chains has at least one flat-state.

Nadia Benbernou and Joseph O'Rourke used cones among other things in their proofs. [19] If a shape is created based on all the positions that the last edge of a fixed-angle chain, where all the dihedral joints, except the last one, are fixed, can take, the result is a cone. The use of cones is a useful tool.

Michael Soss and Godfried Toussaint demonstrated that determining if a given fixed-angle equilateral open chain is flattenable is weakly NP-hard. [35]

Erik Demaine and Sarah Eisenstat extended this result and proved that flattening fixed-angle equilateral chains is strongly NP-hard. For that they used rectilinear planar monotone 3-SAT and semi-rigid chains. A semi-rigid chain is a chain where some dihedral angles are fixed and others not. [16]

From that follows that finding fixed-angle equilateral equiangular obtuse chains that are non-flattenable is at least strongly NP-hard and this in turn means that the detection of locks for these chains is also at least strongly NP-hard.

Erik Demaine, Stefan Langerman and Joseph O'Rourke have worked on the topic of producible polygonal protein chains. They are using fixed-angle equilateral equiangular obtuse open polygonal chains to model the backbones of proteins. The production of proteins by ribosomes they model using a simplified ribosome model, where the chain is produced inside a cone at its tip. On the one hand, they show that only chains that are flattenable can be produced. On the other hand, they show that if there is a non-flattenable configuration for a given α, then the probability of a random configuration being non-flattenable increases with n. For $n \to \infty$, then the probability that a random configuration is non-flattenable approaches one.

[13]

Stefan Langerman introduces the idea, to enlarge an edge by ε to lock special cases. Non-obtuse locks are found on that way with $n = 4$ and $\alpha = 60°$ or $n = 6$ and $\alpha = 90°$.

[2]

2.6. Robot Arm Reachability, Motion Planning and Configuration Space

Robot arm reachability is the question whether the degrees of freedom of a complex body based on an open polygonal chain with fixed start point offer the possibility to move the end point of this chain to a target coordinate without collision in a space with or without obstacles, where the complex body itself can be an obstacle. Motion planning means that the current and target coordinates are known, and a motion is calculated that connects start and target without touching any obstacles. In this context, each joint (fixed joints are not considered as joints in this context) corresponds to a degree of freedom (DOF). Joints of the chain have a local configuration space that is based on an angle between 0° and 360°, or less if limited. Erik Demeine [36] [37], Joseph O'Rourke [36] [37], Lydia Kavraki [38] [39] and Steven LaValle [38] [39] describe this in a more general way, however, universal joints do not interest us, we care about 3D, and the brief summary will suffice.

The configuration space is considered by Erik Demaine and Joseph O'Rourke to consist of all possible coordinates of all vertices of an open polygonal chain. The configuration space is therefore a subset of $\mathbb{R}^{3(n)}$. In general, all configurations of a linkage are its configuration space.

[36]

Lydia Kavraki and Steven LaValle interpret the configuration space, also called C-space, in addition as a torus. [38] [39] To study the configuration space of chains, we will look at the configuration space like Lydia Kavraki and Steven LaValle and extend this concept. Essentially, we don't really care how the coordinates of a chain lie in 3D space. We only care, that adjacent configurations remain adjacent, and collisions can be detected (for collision detection coordinates are necessary). Therefore, we imagine the configuration space, as the set of all joint settings. This reduces the configuration space to a subset of $\mathbb{R}^{(n-3)}$, to be exact $360°^{(n-3)}$ for open chains without limited joints and with intersections allowed. If intersections are prohibited, the space is reduced by the configurations in which intersections are detected. If we now imagine an open chain with five edges and three dihedral joints, we can model the configuration space as a cube. Every configuration is a point in that cubic space. The x-axis represents the first joint, the y-axis the second and the z-axis the third. It is important that the cube cannot be left, because if, for example, you move along an axis in a fixed direction and reach the boundary of the cube and move on, you re-enter the cube on the opposite side. This property allows the cube to represent a torus. To investigate reachability, this form of presentation is very useful, which is based on ideas of my previous research project. [40]

Deborah A. Joseph and W. Harry Plantinga showed that the reachability and motion planning problem for 2D polygonal chains with universal joints is PSPACE complete. [41] Even if these are not 3D fixed-angle polygonal chains, interesting connections can be found between 2D chains with universal joints and these, because dihedral joints in 3D also have a usable angular range of 360°. To answer whether the configuration space of a 3D fixed-angle polygonal chain is connected will therefore probably also be PSPACE complete.

John Canny developed the Probabilistic Roadmap (PRM) in 1987. [42] PRM is a motion planning algorithm, roughly consisting of the following steps: The configuration space is randomly scanned for collisions. Silhouettes are formed around collisions. The determined points outside the silhouettes are connected to form a graph. Finally, shortest paths are determined for this graph, which enables route planning. John Canny proved that his algorithm is in PSPACE.

[37] [38] [39]

George Collins discovered an algorithm in 1973 that can be used for motion planning, which is called Cylindrical Algebraic Decomposition (CAD). [43] To give a brief overview Erik Demaine and Joseph O'Rourke are cited including definitions for the cited part:

"k, the number of degrees of freedom of the mechanism. This is the number of parameters necessary to fully specify the configuration of the mechanism, so that a configuration can be represented by a point in $\mathbb{R}^k$.

m, the number of 'constraint surfaces', each recording some distance or nonpenetration constraint, and each represented by a collection of polynomial equalities and inequalities.

d, the maximum algebraic (polynomial) degree of the constraint surfaces. …

A semialgebraic set is a subset of $\mathbb{R}^k$defined by a Boolean combination of a collection F of polynomial equations and inequalities. A cylindrical algebraic decomposition (CAD) for F is a decomposition of $\mathbb{R}^k$into finitely many cells, which have the property that each polynomial in F evaluates to the same sign $\{-, 0, +\}$ for every point in the cell. For univariate polynomials, a CAD is realized by a partition of $\mathbb{R}^1$ at the real roots of the polynomials. A CAD for multivariate polynomials

is achieved by Collins' recursive algorithm by partitioning each cylinder over the lower-dimensional cells into sign-invariant sectors. The recursion leads to a doubly exponential number of cells: $O((md)3k)$. With cell adjacency stored in a connectivity graph, a motion planning problem can be solved by searching for a collision-free path through this graph between the cells containing the initial and final configurations of the mechanism, in 'randomized expected time' proportional to the number of cells." [4]

We will not use PRM or CAD but take ideas that are contained in these algorithms.

[4] E. D. Demaine and J. O'Rourke, "General Algorithms and Upper Bounds, Probabilistic Roadmaps", in *Geometric Folding Algorithms: Linkages, Origami, Polyhedra*, Cambridge University Press, 2007, pp. 17-19. [37]

3. Unity

The 3D engine of Unity version 2020.3.13f1 is used for model creations and physics simulations. The programming language of Unity is based on C#. A practical advantage of Unity is that adjustments can be tested during runtime. A model can be created without physical properties and transformed into a model with physical properties and vice versa. This means that a model is first created only by placing massless edges in an environment without gravity and collisions. Then mass can be added to this model afterwards, joints can be added and then gravity and collisions can be activated. Fixed joints are used at α-angles and hinge joints with y-anchor and local y-axis are used at φ-angles, which will be called dihedral joints. It was found that all joints in Unity can break under too much load, even when break force and break torque are set to infinity. This happens especially when mass and speed are high. Joints in Unity are not absolutely fixed. This means that if a joint is heavily loaded, it will yield slightly before it breaks, as if you had ordered a steel joint but received one made of rubber. This is also the case when the break force and break torque are set to infinity. However, this is probably used to simulate force transfers in collisions. Another problem is that colliders tend to make errors on very thin objects. Therefore, only edges with radius are simulated. The best possible parameters were sought to reduce the problems. The following settings proved to be very effective:

- Fixed Timestep = 0.0005
- Time Scale = 0.08 (for $m = 600$, can be increased for smaller m)
- Maximum Allowed Timestep = 0.3333333
- Maximum Particle Timestep = 0.03
- Edges:
  - Rigid body:
    - Mass = 1e-07
    - Drag = 0.05
    - Angular Drag = 0.05
  - Radius $\geq$ 0.025 (for Edge Length = 2 or Half Edge Length = 1)
- Joints:
  - Break Force = Infinity
  - Break Torque = Infinity
  - Mass Scale = 1e-05
  - Connected Mass Scale = 1e-05

With these settings, everything is simulated extreme slowly. Joints behave correctly if edges are placed correct. If an edge is placed in such a way that it cannot be straight and therefore the associated joint is slightly bent, this is clearly visible. Collisions are detected correctly. However, if the physics simulation is overloaded anyway, the model explodes instead of crossing edges without collision. Therefore, problems can be clearly detected, and the simulation fulfills its purpose.

4. Impact of Radius

Because in the following models are used to draw conclusions about chains, it is necessary to talk about the influence of radius on edges of a chain or model. Therefore, we start with Lemmas 1 and 2.

Lemma 1:

Radius on edges only has an influence if two non-consecutive edges of a chain or model can touch each other.

Proof:

If no non-consecutive edges of a chain or model can touch, all φ-angles are free. No edge of the chain can be an obstacle for the chain itself.

Lemma 2:

The smaller the radius of edges of a model, the larger the configuration space, as long radius matters.

Proof:

Imagine a model that is in a configuration where the first edge has an intersection with the last edge. When all dihedral joints except the first are fixed for example, it can be clearly seen that by decreasing the radius, the number of non-self-intersecting angles of the first dihedral joint increases. The same applies to any other configuration in which any two edges can touch. The more edges can touch, the more the configuration space is increased, by reducing the radius. There is a clear correlation between the size of the configuration space and the radius of the edges. Based on Lemma 1 this only matters if any non-consecutive edges can touch each other.

5. Closed Chains

Question 1:

Which closed chains are locked? classify locked closed chains.

5.1. Regular Polygons

Question 2:

Which chains can be closed?

A simple class of closed chains are regular polygons with $m > 4$. A regular polygon has one possible configuration in 3D. Even if the configuration cannot be changed, the configuration space cannot be separated. Therefore, regular polygons are not locked. To reproduce this, let us imagine an arbitrary regular polygon. If we rotate all dihedral joints of this polygon by π, we obtain a new regular polygon with reversed edge order. However, if we then rotate the entire new regular polygon by π, we get the original regular polygon with the original edge order again. Therefore, we say that the configuration in which all dihedral joints are set to zero and the one in which all dihedral joints are set to π are the same for regular polygons. If the dihedral angles were set differently, the chain cannot be closed. If we increase the α-angle of a regular polygon, this chain cannot be closed even if all dihedral angles are zero or π. This results in Proposition 1 and 2. In the following we will try to give a more precise answer to Question 2.

Proposition 1:

All chains with $\alpha(m) = \frac{\pi \cdot (m-2)}{m}$ are rigid and not locked.

Proposition 2:

All chains with $\alpha(m) > \frac{\pi \cdot (m-2)}{m}$ cannot be closed.

5.2. Locked Closed Chains

Question 3:

Which closed chains can be used to tie non-trivial knots?

Question 4:

Which closed chains can have a self-intersection?

Suppose we have two identical closed ropes, one rope in an unknot configuration and the other in a trefoil knot configuration. Both ropes are in different configuration subspaces. It is only possible to transfer the first rope into the configuration of the second rope if the rope is cut and rejoined. The same applies to any two different knots. Figure 4 shows three separated configuration-subspaces of the same closed rope.

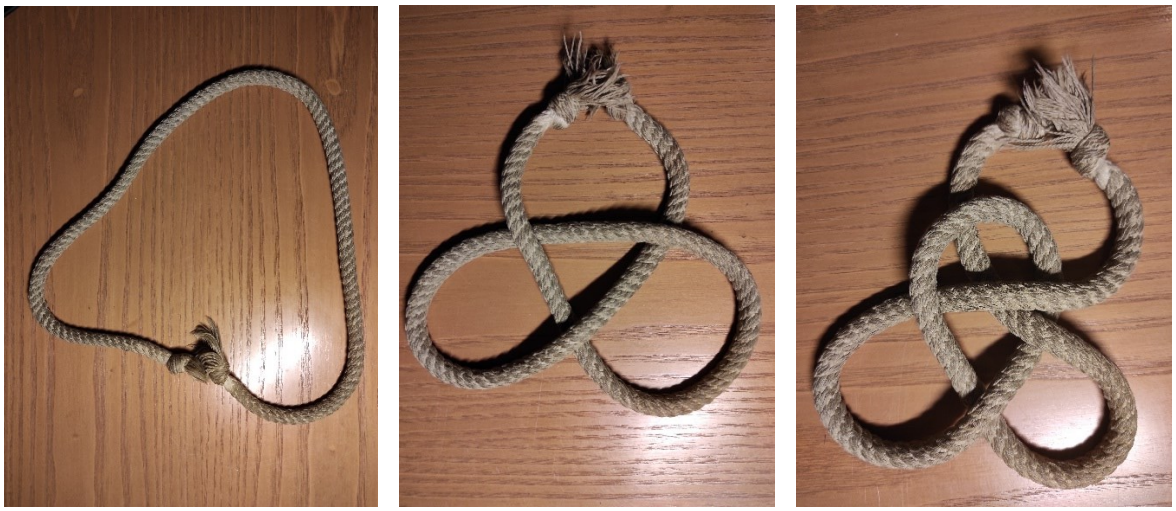

Figure 4: Three separated configuration-subspaces, unknotted, and two separated prime knots, 3_1*, also called trefoil knot, and* 6_2*.*

Even if each rope can be understood as a chain with many short edges, it is still not clear if the beginning and the end can meet exactly. Therefore, an unknot model was constructed and a trefoil knot model, both with equal number of edges and equal α-angle. The α-angle is chosen so that three consecutive edges can form a 90° angle and several edges in the zig-zag pattern can form a straight line from the macro perspective. Therefore, chains based on $\alpha = 135°$ can be closed easily if only turn and zig-zag patterns are used, and all dihedral angles are 0°, 90°, 180°, or 270°. Both models are shown in Figure 5.

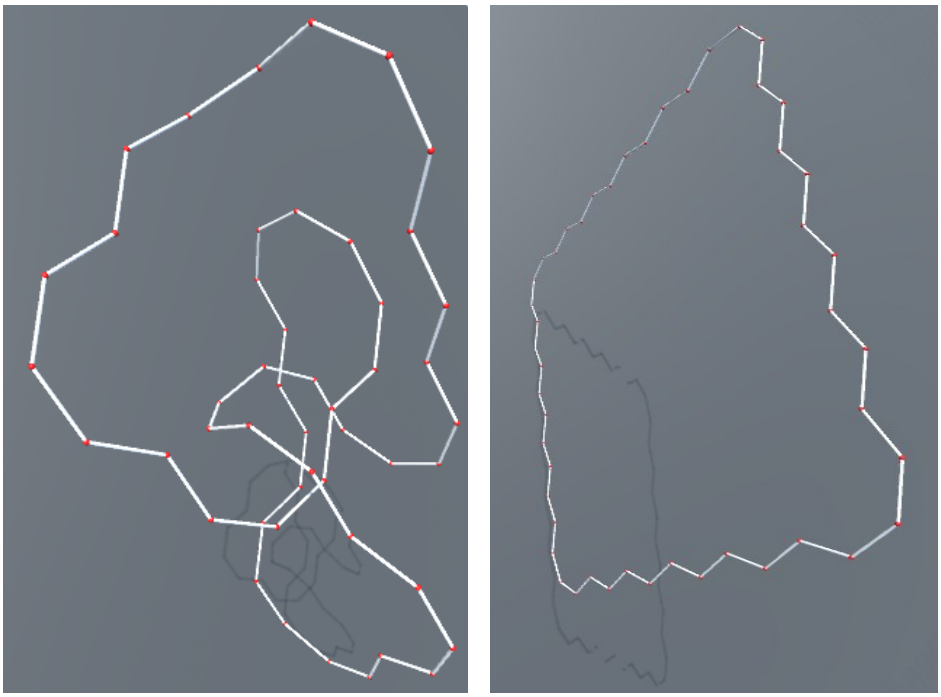

Figure 5: Trefoil and unknotted closed models with m=48, l=2, r=0.1, and α=135°.

If the trefoil knot from Figure 4 is placed slightly differently, it becomes clear that it consists of two loops and ε to prevent crossings and to arrange it in 3D. Figure 6 shows that knot again. The Fary-Milnor Theorem tells us that the curvature of any smooth non-trivial Knot is greater than 4π. If we now combine this with the total curvature definition from Kenneth C. Millet and remember that our chains must be obtuse, it follows that at least $m > 8$ is necessary to form a closed chain with total curvature greater than 4π.

Theorem 1 (Fary-Milnor):

The curvature of any smooth non-trivial knot is greater than 4π.

[3]

Definition (simplification of Kenneth C. Millet):

The total curvature of a closed chain equals $n \cdot (\pi - \alpha)$.

[30]

Figure 6: Smallest knot

In simple examples we have seen how chains can be closed. To close a chain not only the end node of the last edge but also its start node must be placed correctly. This causes the same problem between the second to last and the last edge and so on. Algorithm 1 was developed to get over this problem. For that, Algorithm 1 approximates all dihedral angles of a closed chain, starting from the configuration of an open chain with two additional edges. Here $dist(a, b)$ calculates the Euclidean distance between a and b, and $random(x)$ returns a random value in range from $-x$ to x. For Algorithm 1 it is necessary that self-intersection is allowed, but mainly chains are produced which do not have self-intersections. Surprisingly, this works not only for very small chains, such as those with $n = 6$ where the algorithm needs some seconds for solutions with an error in the fifth decimal place, but also for chains with many more edges. For example, a chain with $m = 100$ can be closed in some minutes, with the same error. Algorithm 1 is also applicable to chains with odd and even edge counts. In rare cases, Algorithm 1 may not terminate if a local minimum of the objective function is found that cannot be closed. If particularly large chains are to be closed, a start configuration with only one additional edge, as well as an adapted target function, can also be used. This approximates a valid closed chain, but then the output lacks a dihedral angle, which would have to be calculated alternatively. An advantage on the other hand is that less local minima occur in that case. If large chains are closed, it may be necessary to adjust the decay section of the algorithm.

Two simple start configurations lead to useful results, apart from random configurations, when using Algorithm 1: the all-zero configuration in which all dihedral angles are zero and the zig-zag configuration in which dihedral angles alternate between zero and π. In most cases, Algorithm 1 produces a knot or a configuration which has no simple projection from an all-zero configuration. For this at least as many edges are needed that two circles with the given α-angle can be formed. If two circles cannot be formed, Algorithm 1 produces approximately the same results with an all-zero start configuration as with a zig-zag start configuration or is caught in a local minimum. A zig-zag start configuration is used to create a configuration with a simple projection, a shape that can usually resemble the outline of a potato chip. Figure 7 shows three closed models created with Algorithm 1. For the left two models we started with the all-zero configuration and for the right one with the zig-zag configuration. All three use $m = 9$ and $\alpha = 95°$, thus two circles can be formed. After testing the models in a physics simulation with a vibrating plate it was determined that all three models are in disconnect configuration subspaces. After rebuilding the left two models as rope models it turned out that the left model corresponds to the smallest knot and the middle model can be unfolded (as a rope model) to a configuration with a simple projection.

Algorithm 1: Probabilistic 3D Fixed-Angle Chain Closing

Input: Φ_{open+2}, α, ε<0.0099

Output: Φ_{closed}

step = 0, range = 0.99

$\Phi_{closed} = \Phi_{open+2}$

create c_{open+2} based on Φ_{open+2} and α

score = dist(v_1, v_{m-1}) + dist(v_2, v_m) + dist(v_3,v_{m+1}) of c_{open+2}

While *score* > *ε:*

Foreach φ in Φ_{open+2}*:*

$$\varphi = \varphi + random(range)$$

update c_{open+2} with Φ_{open+2}

score2 = dist(v_1, v_{m-1}) + dist(v_2, v_m) + dist(v_3,v_{m+1}) of c_{open+2}

If *score2* < *score:*

score = score2

$\Phi_{closed} = \Phi_{open+2}$

Else*:*

$\Phi_{open+2} = \Phi_{closed}$

If *score* > *ε:*

step++

If *mod(step, 1000) == 0:*

range = range · range + ε

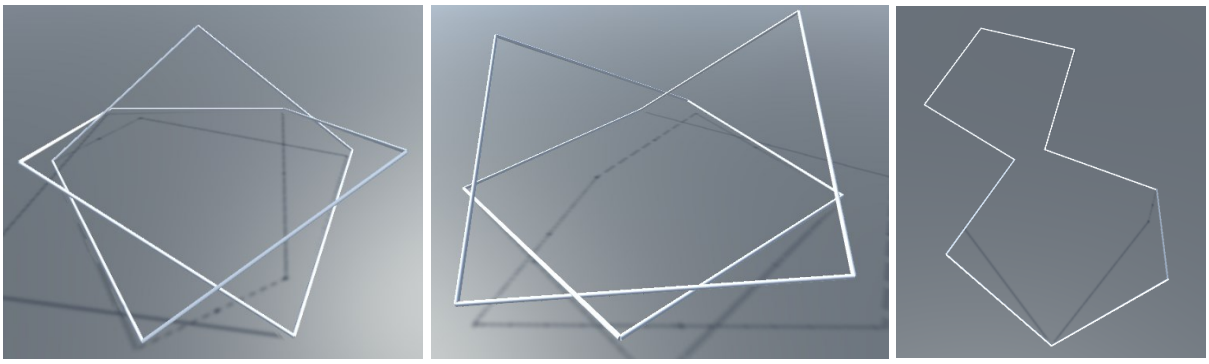

Figure 7: Three physics models generated using Algorithm 1 with l=2, r=0.025, m=9, and α=95°. The two left models started with an all-zero and the right with zig-zag configuration.

Now we are looking for a definition of locked closed chains that can form non-trivial knots and unknots based on two circles. Lemma 2 is considered, which is why formula statements apply to chains with infinitesimal edge radius.

The equation for α-angles of regular polygons is:

$$\alpha_{regular\ polygon}(m) = \frac{\pi \cdot (m-2)}{m} \tag{1}$$

We combine this equation with the number of loops we need for a knot and simplify it:

$$\alpha_{2circles}(m) = \frac{2\pi \cdot \left(\frac{m}{2}-2\right)}{m} \tag{2}$$

Because we need additional ε and search for a threshold we rewrite the equation as an inequality:

$$\alpha_{knot}(m) < \frac{2\pi \cdot \left(\frac{m}{2}-2\right)}{m} \tag{3}$$

Let's try with $m = 8$ which should lead to an orthogonal angle:

$$\alpha_{knot}(8) < \frac{360° \cdot \left(\frac{8}{2}-2\right)}{8} = 90° \tag{4}$$

Because α must be obtuse we try $m = 9$:

$$\alpha_{knot}(9) < \frac{360° \cdot \left(\frac{9}{2}-2\right)}{9} = 100° \tag{5}$$

Now we check the result with $m = 13$:

$$\alpha_{knot}(13) < \frac{360° \cdot \left(\frac{13}{2}-2\right)}{13} \approx 124.62° \tag{6}$$

To check the result, two new models are created with Algorithm 1 that are shown in Figure 8. Both with $m = 13$, the left one with $\alpha = 124°$ and the right one with $\alpha = 125°$. The left one has no simple

projection and can be created with all-zero start configuration. The right one has a simple projection but cannot be created from the all-zero start configuration because only not closable local minima are targeted by Algorithm 1 in that case. Therefore, the right one is generated with zig-zag start configuration. By testing in physics simulation, it is determined that the left model cannot be unfolded into a model with simple projection and that the right one cannot form any self-intersection even if this would be allowed. This leads to Corollary 1.

Corollary 1 of Theorem 1:

All closed chains with $\alpha(m) < \frac{2\pi \cdot \left(\frac{m}{2}-2\right)}{m}$ can form a non-trivial knot.

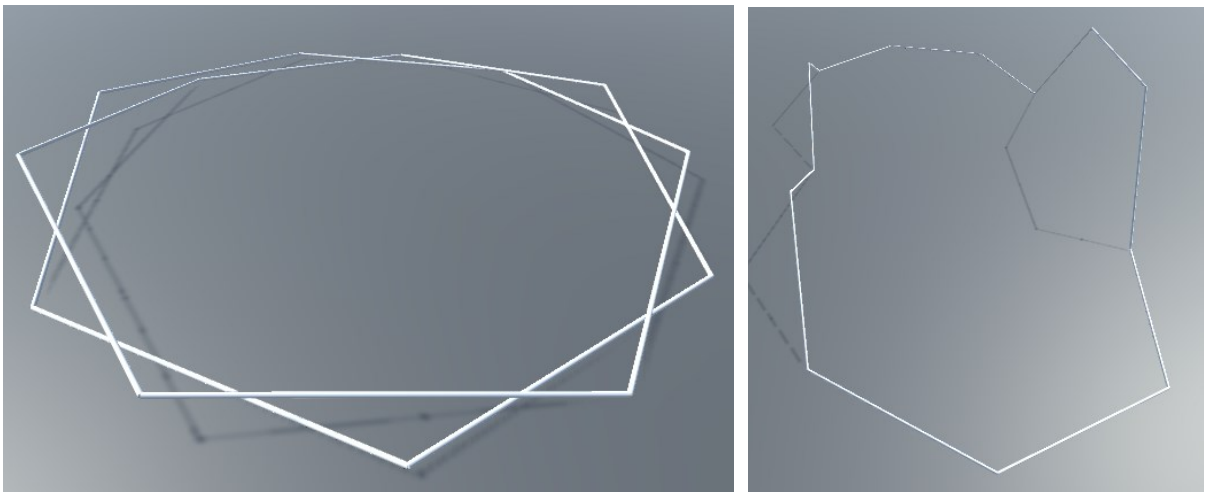

Figure 8: Two physics models generated using Algorithm 1 with l=2, r=0.025, m=13, the left has α=125° and the right α=124°

In Figure 9, the same α-angle is used as in Figure 7, but the number of edges is reduced. The range $4 < m < 9$ is examined. The chain with $m = 5$ could not be closed in tests using zig-zag, all-zero and random start configurations and therefore that chain is not included. Based on observations during physics simulations with these models, it has been noticed that closed chains with $m < 9$ cannot form self-intersections. Another observation is that the models in Figure 9 are not rigid.

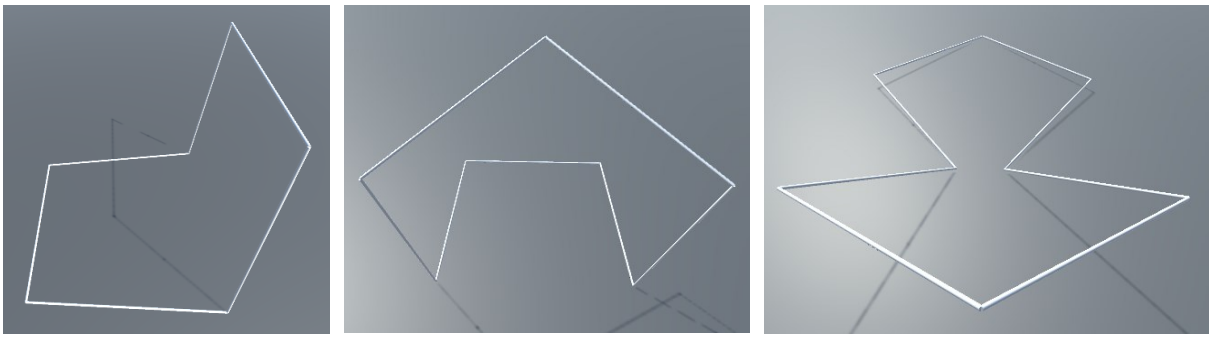

Figure 9: Three physics models generated using Algorithm 1 with l=2, r=0.025, and α=95°. The left model with m=6, the middle with m=7, and the right with m=8.

The same is observed for any closed chain with total curvature below 4π. It follows that all closed chains that cannot form two circles cannot form a self-intersection, even if self-intersection is allowed. This results in Conjecture 1, Open Problem 1.

Conjecture 1:

All closed chains with $\alpha(m) > \frac{2\pi \cdot \left(\frac{m}{2}-2\right)}{m}$ cannot form self-intersections, even if this would be allowed.

Open Problem 1:

Prove or disprove Conjecture 1.

To show that probably all closed chains are locked with $m > 5$, which are not contained in $\alpha_{knot}(m)$ or $\alpha_{regular\ polygon}(m)$, the configuration spaces of the chains of Figure 8 are examined. In physics simulations the chains have shown a cycle of movements. Essentially, this means that the locale configuration subspace is based on a circle and for high n a circle with attachments or volume. Then a wooden model of a closed chain with $m = 8$ was created and the previously observed cyclic motion was performed there. Two opposite nodes of the chain were marked, v_1 and v_5. For both nodes, one side was marked white and the other side black. Side means in this context that if the two adjacent edges of a node form a 'V' from our perspective, we can see one side and if this 'V' is rotated 180° in 3D so that we see a 'V' again, we see the other side. Eight distinctive configurations, which look equal, with equal spacing on the circle of the configuration subspace are now considered. Apart from these eight configurations, there are no others in the configuration subspace, which look exactly like this. In Figure 10, the first four configurations are shown, the other four correspond to a rotation of the images by 180° and are therefore omitted. The configurations are presented in such a way that they can be easily compared, that means that the entire model is rotated each time. It can be clearly seen that for both marked nodes only two of the eight configurations show the white side up. It follows from this that there must be a second configuration subspace which represents a mirror image of the configuration subspace examined, where, with the same representation by the model, two of the eight configurations represent the two marked nodes with the black side up. The same is found for closed chains with $m = 6$. For chains with an even number of edges, which are non-self-intersectable, and have $m > 8$, similar results can be observed, but the configuration subspaces are larger and therefore it is only assumed that there are at least two disconnected configuration subspaces which are mirrored from each other. On a sample basis, other angles in $\alpha_{non-self-intersectable}(m)$ were also checked and gave the same results. This leads to Conjecture 2 and Open Problem 2.

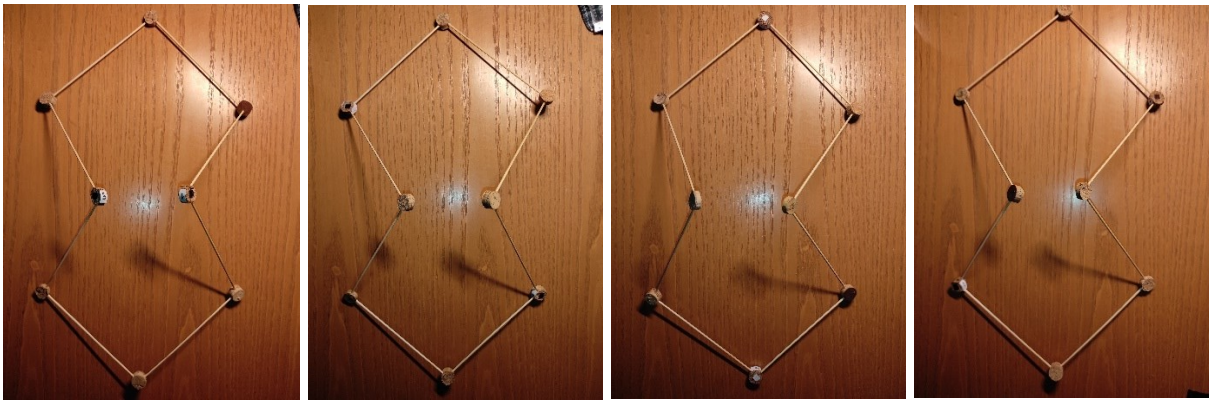

Figure 10: First half of eight distinctive configurations with equal spacing on the circle of the configuration subspace of a wooden model with m=8, l=19, r=0.15, and α≈95°.

Conjecture 2:

All closed chains with $\frac{2\pi\cdot\left(\frac{m}{2}-2\right)}{m} < \alpha(m) < \frac{\pi\cdot(m-2)}{m}$ and even m are locked.

Open Problem 2:

Prove or disprove Conjecture 2.

In Figure 10, the marked nodes move once through each possible node position of the plot within a cycle, with each node position always pointing up the same side when the same node index returns to that position. The question that now arises is what happens with an odd number of nodes. Therefore, the experiment is repeated with a closed chain with $m = 7$. It is found that with an odd number of edges, a cycle contains twice as many configurations that look the same, with each node position occurring once with white side up and once with black side up. Therefore, only one configuration subspace is found on this way and not two separated as in the case of chains with an even number of edges. Therefore, it is tried to find a configuration which is not in the cycle of the examined configuration subspace. By using Algorithm 1 with random start configurations, a second configuration subspace is identified. Figure 11 shows a comparable configuration from both configuration subspaces. Even if a test of higher edge numbers is again omitted, the same behavior is expected. On a sample basis, other angles in $\alpha_{non-self-intersectable}(m)$ were also checked and gave the same results. This leads to Conjecture 3 and Open Problem 3.

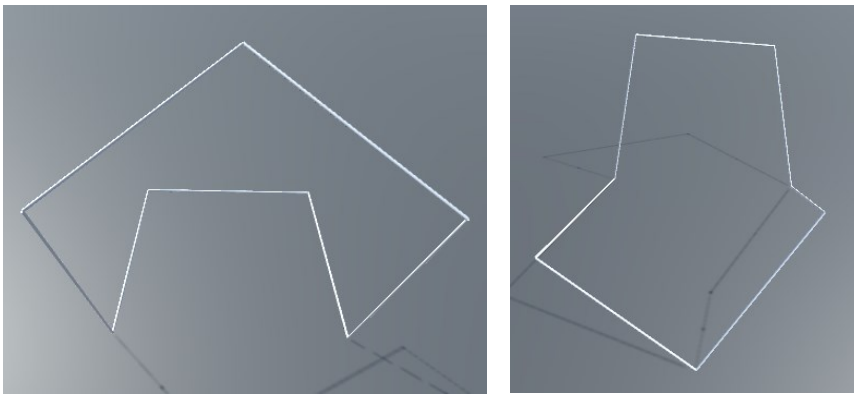

Figure 11: Two physics models in separated configuration cycles with l=2, r=0.025, m=7, and α=95°.

Conjecture 3:

All closed chains with $\frac{2\pi\cdot\left(\frac{m}{2}-2\right)}{m} < \alpha(m) < \frac{\pi\cdot(m-2)}{m}$ and odd m are locked.

Open Problem 3:

Prove or disprove Conjecture 3.

If Algorithm 1 is slightly modified, chains with even number of edges can be created in a special configuration. Here, start configurations are used where the dihedral angles are alternately 90° and

270°. All even joint numbers are changed with the same random value and all odd ones with a different random value, which is the same for all odd joint numbers. This results in a valid closed chain with a regular pattern. An interesting observation is that for $m = 6$ all possible α-angles in this configuration lead to a rigid closed chain. This leads to Conjecture 4 and Open Problem 4. For all other edge counts except $m = 5$, these configurations are contained in the configuration subspaces described earlier. Figure 12 shows a rigid chain with $m = 6$. As a result, the configuration space of closed chains with $m = 6$ consists of at least four configuration subspaces, two of which are rigid and two of which are based on a circle.

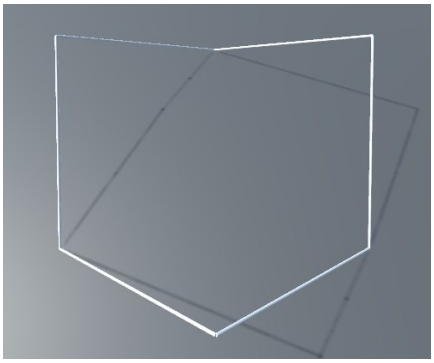

Figure 12: Rigid closed physics model with l=2, r=0.025, m=6, and α=95°.

Conjecture 4:

All closed chains with $m = 6$ and $\frac{\pi}{2} < \alpha < \frac{2\pi}{3}$ have two rigid configurations $\Phi_1 = (\beta, \gamma, \beta, \gamma, \beta, \gamma)$ and $\Phi_2 = (\gamma, \beta, \gamma, \beta, \gamma, \beta)$, where $\beta \neq \gamma$ and, β and γ are any angles, for which this chain is closed.

Open Problem 4:

Prove or disprove Conjecture 4.

All examined chains with $\alpha(m) < \alpha_{regular\ polygon}(m)$ could be closed, except chains with $m = 5$. This leads to Conjecture 5 and Open Problem 5.

Conjecture 5:

The only closed chain with $m = 5$ is the regular polygon. All chains with $m > 5$ and $\frac{\pi}{2} < \alpha \leq \frac{\pi \cdot (m-2)}{m}$ can be closed.

Open Problem 5:

Prove or disprove Conjecture 5.

5.3. Closed Chains Summary

Closed chains can be divided into subgroups:

$$\alpha_{regular\ polygon}(m) = \frac{\pi \cdot (m-2)}{m} \tag{7}$$

$$\alpha_{2circles}(m) = \frac{2\pi \cdot \left(\frac{m}{2} - 2\right)}{m} \tag{8}$$

$$\frac{\pi}{2} < \alpha_{knot}(m) < \alpha_{2circles}(m) < \alpha_{non-self-intersectable}(m) \leq \alpha_{regular\ polygon}(m) \tag{9}$$

Regular polygons are rigid and not locked.

For $m = 5$ it is found that only the regular polygon can be closed.

For $m = 6$ at least two rigid configurations for any closable α-angle exist, except for regular polygons.

$\alpha_{2circles}(m)$ is the boundary below which closed chains can form a non-trivial knot and above which closed chains could not form a self-intersection if it would be allowed.

$\alpha_{knot}(m)$ does not exist for all closed chains with $m < 9$.

All closed chains with $\alpha_{knot}(m)$, can form a non-trivial knot and the unknot. Therefore, all closed chains with $\alpha_{knot}(m)$ have at least two disconnected configuration subspaces and are locked.

All closed chains with $\alpha_{non-self-intersectable}(m)$ and $5 < m < 9$ have at least two disconnected configuration subspaces and are therefore locked. The non-rigid configuration subspaces of these closed chains with even edge counts are mirror images of each other while for odd edge counts this is not true. These configuration subspaces are based on topological circles with attachments or volume.

For all closed chains with $\alpha_{non-self-intersectable}(m)$ and $m > 8$ it is expected that these have at least two disconnected configuration subspaces and are therefore locked.

For all closed chains with $\alpha_{2circles}(m)$ it is expected that these are locked, because the transition points between the configuration subspaces are blocked by self-intersections.

If the results are combined, Conjecture 6 follows, which can be proved based on solving the open problems of closed chains.

Conjecture 6:

All closed chains with $m > 4$ and $\alpha(m) < \alpha_{regular\ polygon}(m)$ are locked, all others not, but these are rigid.

6. Open Chains

Question 5:

Does a locked open chain exist? If yes, classify locked open chains. This question is asked by Erik Demaine. [1] [2]

6.1. At First Glance

The first idea to find locked open chains, started with wooden models like the one shown in Figure 13. It is based on a knot like or a braiding-based structure, with the hope, that this configuration is non-flattenable.

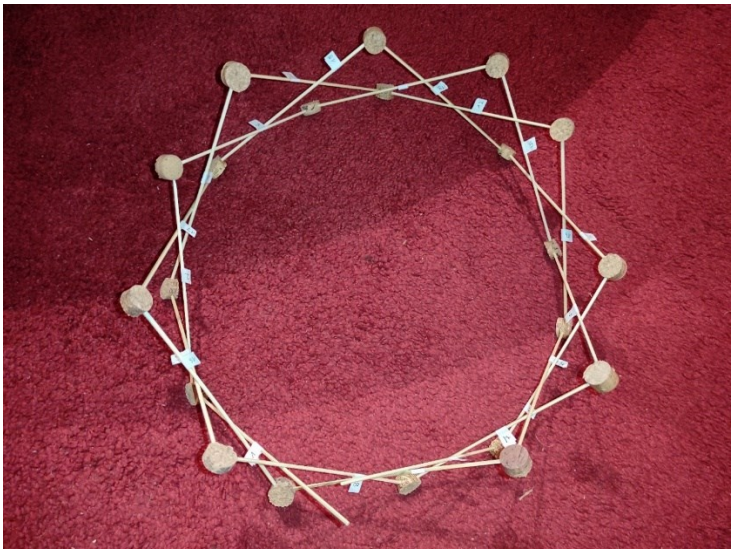

Figure 13: First locked open chain idea as wooden model with m=22, l=19.5, r=0.15 and α≈98°

Because the model is quite stable and immobile in this form, but extremely inaccurately constructed, the question now is whether, this really has all the properties of a chain. The recreation with an algorithm which uses edges with radii led to the result, that some edges are crossing, while others have no contact to near edges. Therefore, the assumption comes up that too much movement is possible to cause a lock if equal infinitesimal radii are used. Figure 14 shows an example which is created with algorithm 2.

Algorithm 2: Intertwined Circular Path

Set every dihedral angle in the following way:

For i in range$(m-2)$:

If $(i \bmod 2 == 0)\ \varphi(i) = \beta + \gamma \cdot (i \bmod 3)$

Else $\varphi(i) = -(\beta + \gamma \cdot (i \bmod 3))$

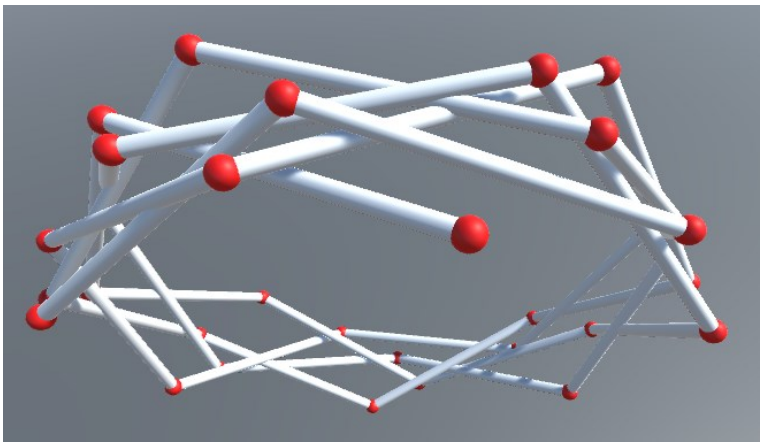

Figure 14: Model based on algorithm 2 with m=30, l=2, r=0.1, α=120°, β=60° and γ=3°

6.2. Rigidity

To determine a lock, it is useful to know the conditions under which a section of a chain is rigid. Basically, there are two different properties whose combination can lead to rigidity. The first property assumes, that the start and end edge of a chain section are fixed. If the chain section is the shortest connection between the start and end edge, the chain section is rigid. The second property is based on edges that are constraint or in other words which are touched by other edges.

If there is an open chain with $m > 2$ for which it holds that all dihedral joints are rigid, it is locked. This is true because every open chain with $m > 2$ has multiple configurations where at least one is flat. The question now is whether the combination of touches and shortest possible chain sections is sufficient to block all dihedral joints.

Therefore, different algorithms have been developed to find an open rigid chain. Essentially, all these algorithms are based on a central point, trying to place all edges with minimum distance to it. This can cause two edges to cross. To prevent this, a check is made before adding an edge to see if it intersects another. If this is the case, the last dihedral angle is adjusted so that this crossing is just avoided. The used tactics are always dodge right, alternate dodge left and right, and dodge so that the distance from the center point is minimal. In addition, these tactics are also used with backtracking. Backtracking means that it is stored which edge is touched last by another edge. The dodge tactics are now applied to all last dihedral joints that are unconstraint. In the process, the angle changes are distributed evenly over all last unconstraint edges.

Another idea is that if all edges are placed with minimal distance to the center, and the first and last edge are touched by other edges, this could lead to an invariance of all dihedral joints. Algorithm 3 summarizes the described in a simple form. Constraint in this context uses so small ε, that only touching edges are considered.

Therefore, models were generated using algorithm 3 until a small model was found, that satisfies the conditions, and collision points were marked. Then this model was rebuilt as a wooden model. It was noted that the chain seemed to be stable on the one hand, but still moveable in one or two places. Both Models are shown in Figure 15.

Algorithm 3: Intertwined Spherical Path

While start and end edge are not full constraint:

    Add an edge in minimum distance to the center point

    If (the added edge intersects another edge):

        If (backtracking is disabled):

            Evade by adjusting the last φ-angle

        Else:

            Evade by adjusting φ-angles of last consecutive unconstraint edges

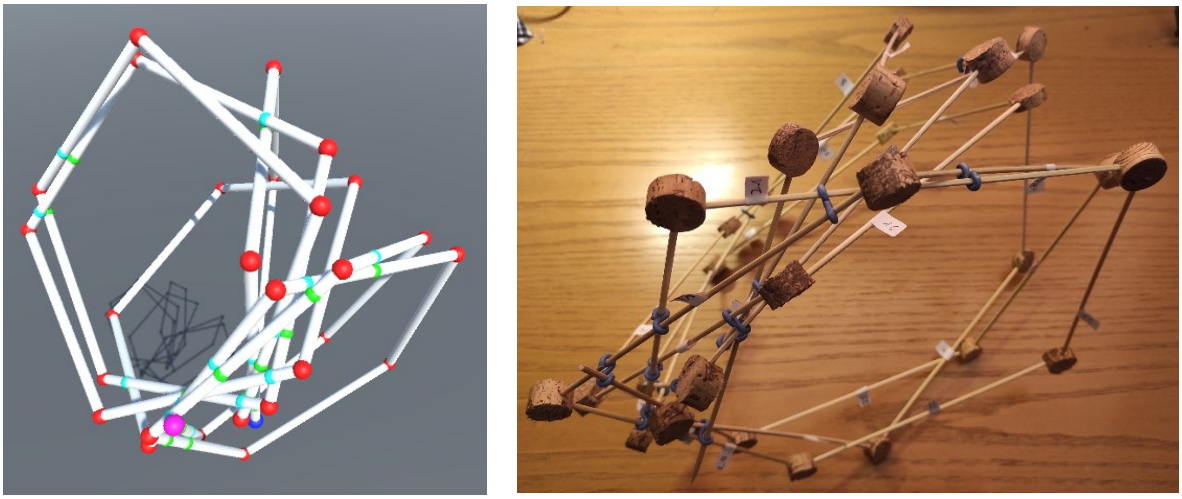

Figure 15: Model with m=29, l=2.04, r=0.1, α=95° and markers and a wooden copy with n=29, l=19.5, r=0.15, α≈95°, and distance holders. Start node is blue, end node is pink, left constraint is teal, and right constraint is green.

To determine, whether these are rigid chains, corresponding models are translated to physics models and are dropped on a ground plate. If the model does not deform when colliding with the ground plate, an invariance on all dihedral joints and thus a lock is detected. When the model of Figure 15 was dropped in this simulation and collided with the ground plate, there were so many movements that it can be ruled out that this model is rigid or non-flattenable. The same was found for all tested chains based on algorithm 3 with $m < 700$. It was found that the edges restricting the start and end edges could be moved. For this idea to work, the constraining edges themselves must also be constrained, and the edges that are used for those constraints must also be constrained, and so on.

Because in rare cases an edge is placed such that no further edge can be added, many attempts have been made to find possibly large open chains that are rigid. To get around this problem, the idea came up that if no more edges can be added, the model is converted to a physics model, then dropped, converted back to the original model after the collision, and then generation is continued. In most cases, this creates the possibility of placing additional edges. Algorithm 4 summarizes the described in a simple form.

Algorithm 4: Extending Intertwined Spherical Path

While dropping enables adding more edges:

    Add an edge in minimum distance to the center point

    If the added edge intersects another edge:

        If backtracking is disabled:

            Try to evade by adjusting the last φ-angle

        Else:

            Try to evade by adjusting φ-angles of last consecutive unconstraint edges

    If there is no valid place for the last edge:

        Remove last edge(s)

        Transform the model into a physics model

        Drop the physics model on a plate

        Transform the physics model back into a model

No rigid open chain was found based on algorithm 4.

6.3. Unfolding Resistance

To determine whether a model that is not rigid can nevertheless not be unfolded, a up and down movement of a ground plate has proven to be extremely effective. In doing so, a model bounces like a ball, spins, rotates, and deforms. In most cases, a release of start or end edge can be detected after a few seconds. In a few cases, this process can take a very long time. The longest release of an end edge took 12 hours. Hundreds of models were found, all of which could be unfolded. Initially, small numbers of edges were tried and then number of edges were increased and tested again. Because of this only the unfolding of start and end edges are of interest. For this a function was used which allows to transform and drop the model and transform it back whenever wanted. However, because tests with small numbers of edges were unsuccessful, larger numbers of edges were also investigated.

Only two models are found on this way, which still have not unfolded start and end edge after 24 hours in physics simulation. Model m_1 with $m = 514$ and $\alpha = 93°$ and model m_2 with $m = 613$ and $\alpha = 113.1°$ which were generated using algorithm 4. These two models were examined in more detail. Figure 16 shows both models.

In the physics simulation, both models have little to no movement on the inside and the outer joints are more flexible. This can also be seen in Figure 17 and 18, where the number of edges with distance of one or less to each edge was calculated. There is a clear correlation between the distance from the central point of the ball and the number of edges in the vicinity. For this distance calculation, the center point of each edge was used instead of nearest point of two edges.

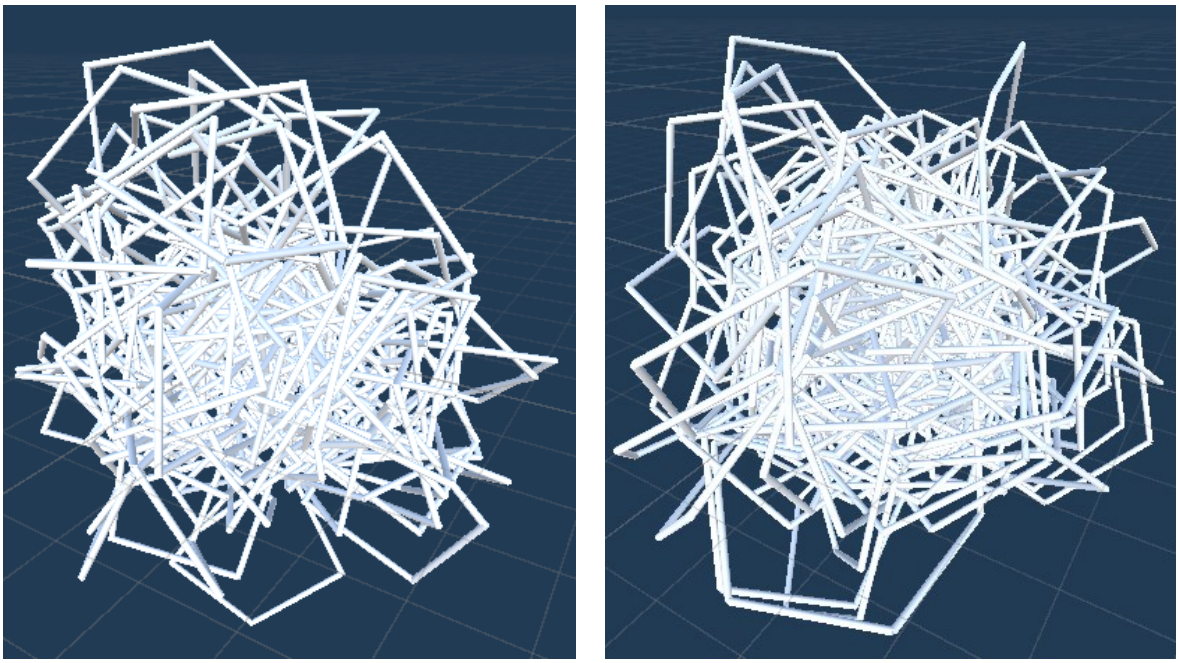

Figure 16: Left m_1 with m=514, α=93°, and right m_2 with m=613, α=113.1°, both with l=2.04 and r=0.1

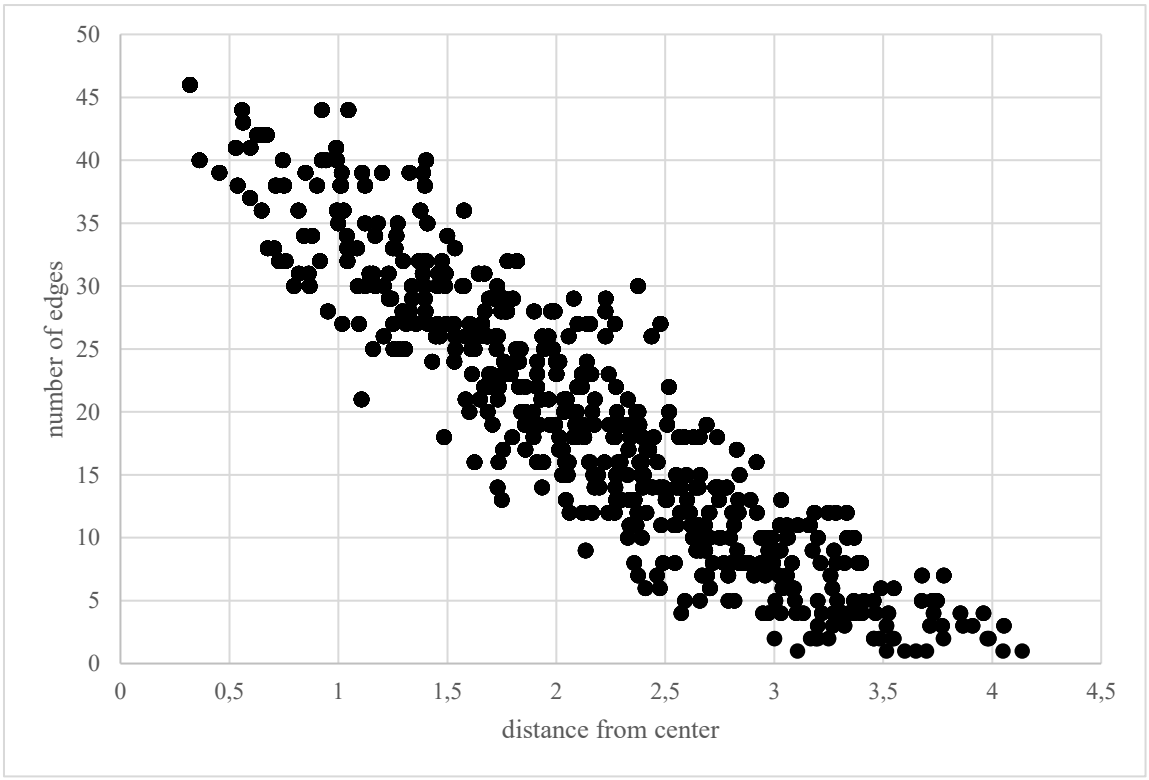

Figure 17: number of edges in distance one or less per distance from center for each edge of m_1 with l=2.04, r=0.1, m=514, and α=93°

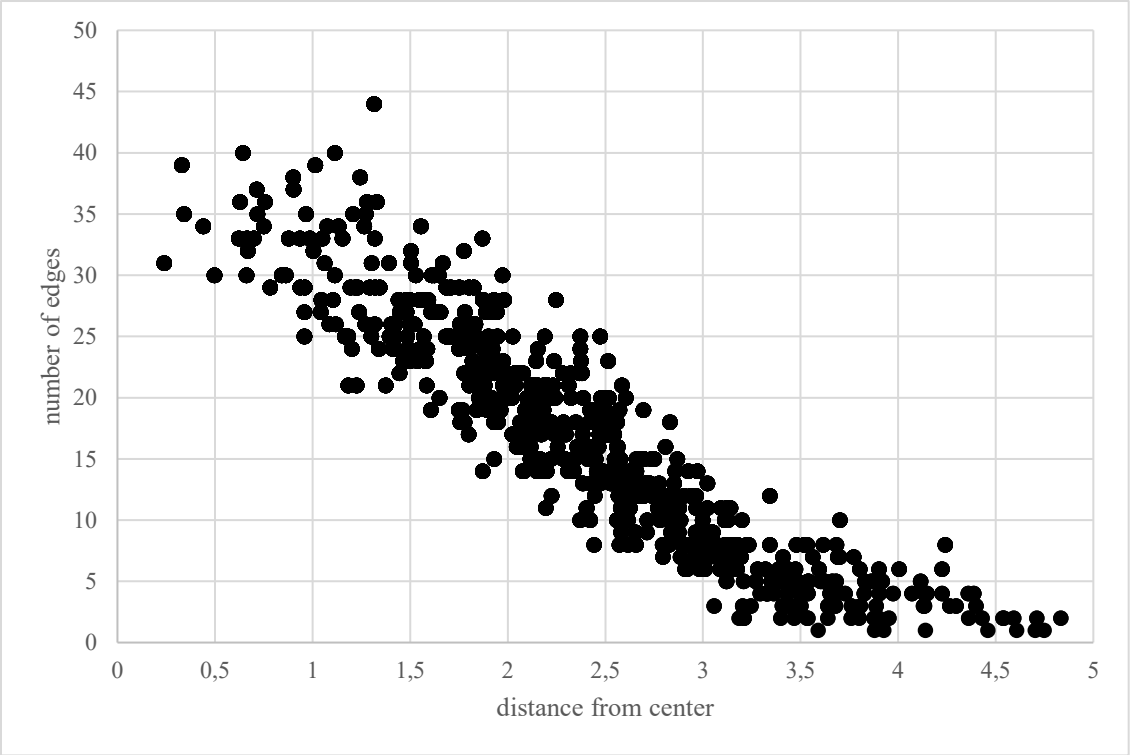

Figure 18: number of edges in distance one or less per distance from center for each edge of m_2 with l=2.04, r=0.1, m=613, and α=113.1°

To understand in which areas sections of the models are located, the distances of all edges to the central point of the model are shown in Figure 19 and 20 as a function of the edge index. There, an increase in the distance to the central point based on the edge index can be detected. Particularly striking are the last edges of m_2, which show a small distance to the central point despite an increasing trend. Because both the edges at the start and end of m_2 are in low-motion regions, this seems to favor a lock or at least extremely unfolding resistance.

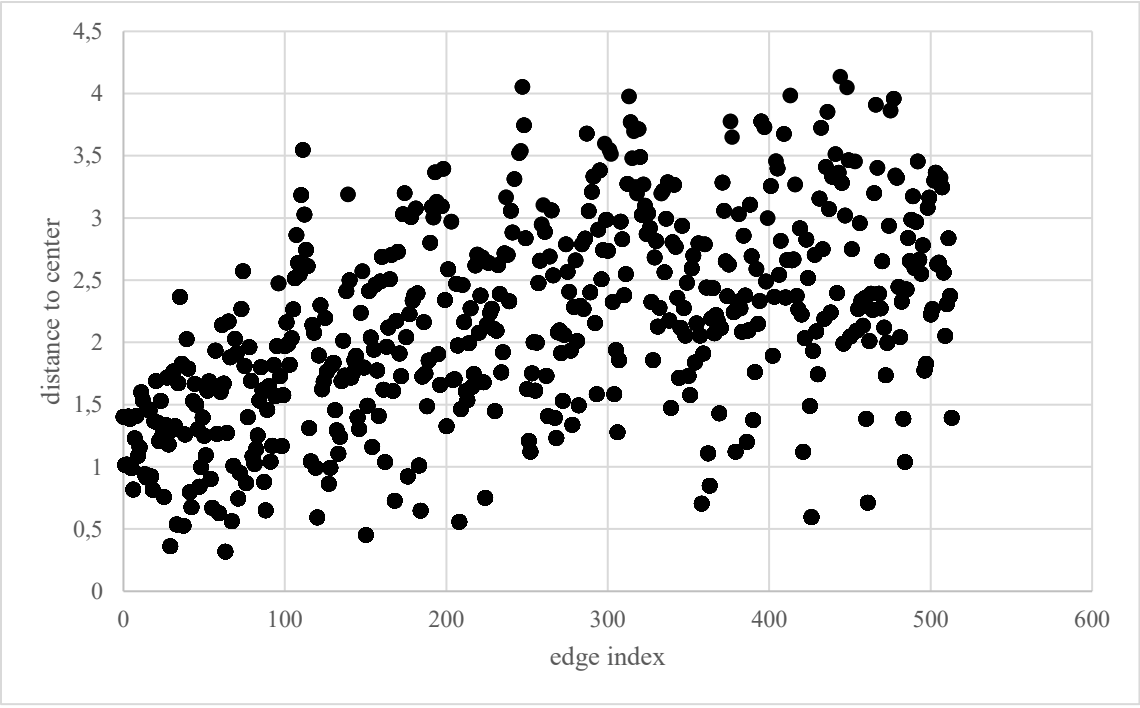

Figure 19: Distance to center per edge index for m_1 with l=2.04, r=0.1, m=514, and α=93°

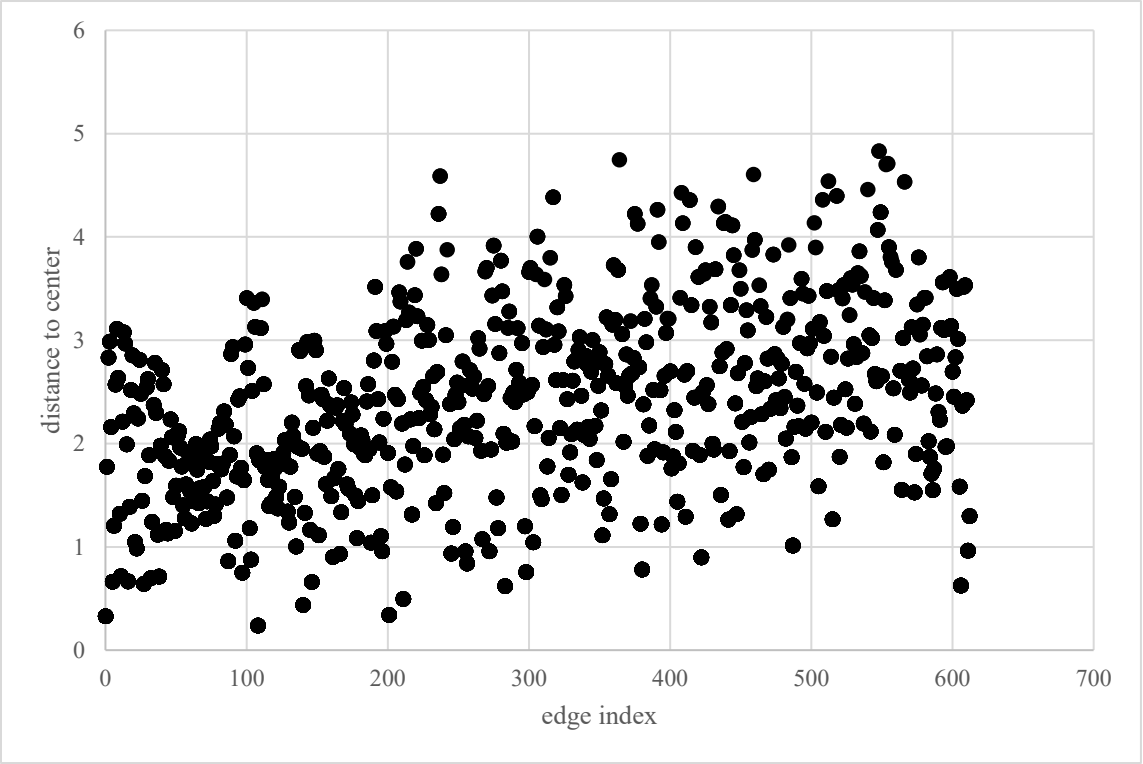

Figure 20: Distance to center per edge index for m_2 with l=2.04, r=0.1, m=613, and α=113.1°

How often edges are placed in which distance can be seen in Figures 21 and 22. Here it is noticeable that most edges do not occur in the center, but on a spherical surface with radius of about 2.2.

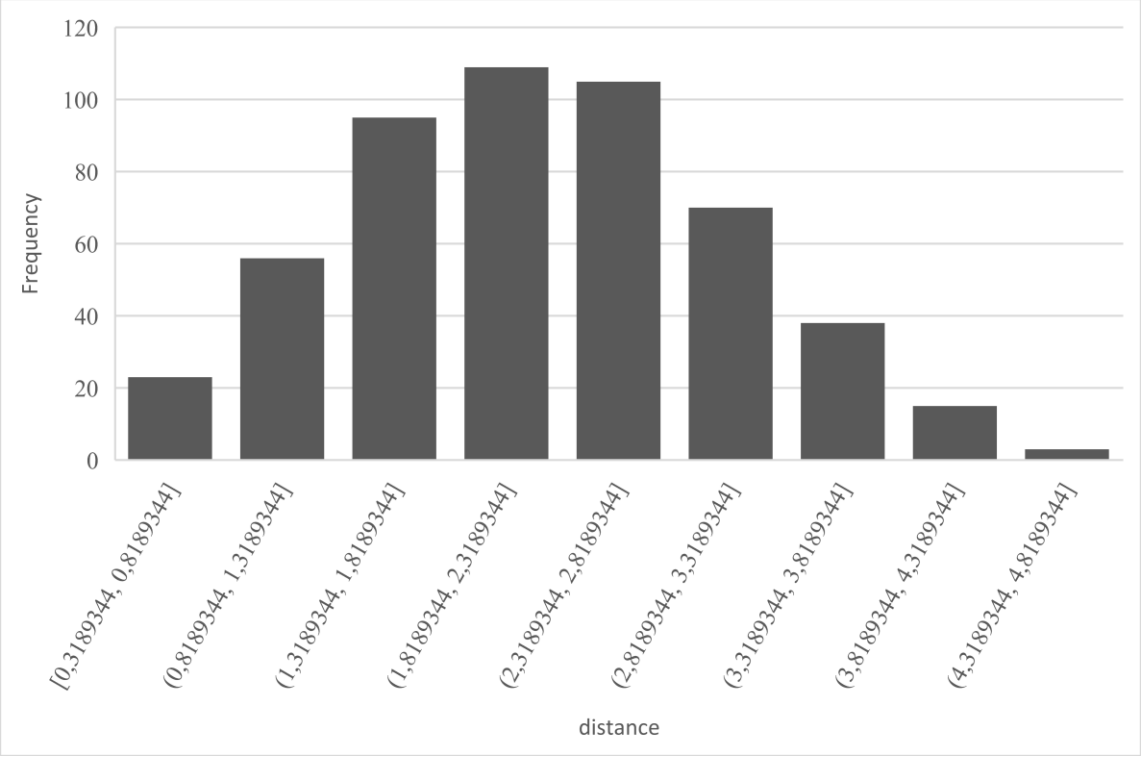

Figure 21: Frequency of distances from center point for edges of m_1 with l=2.04, r=0.1, m=514, and α=93°

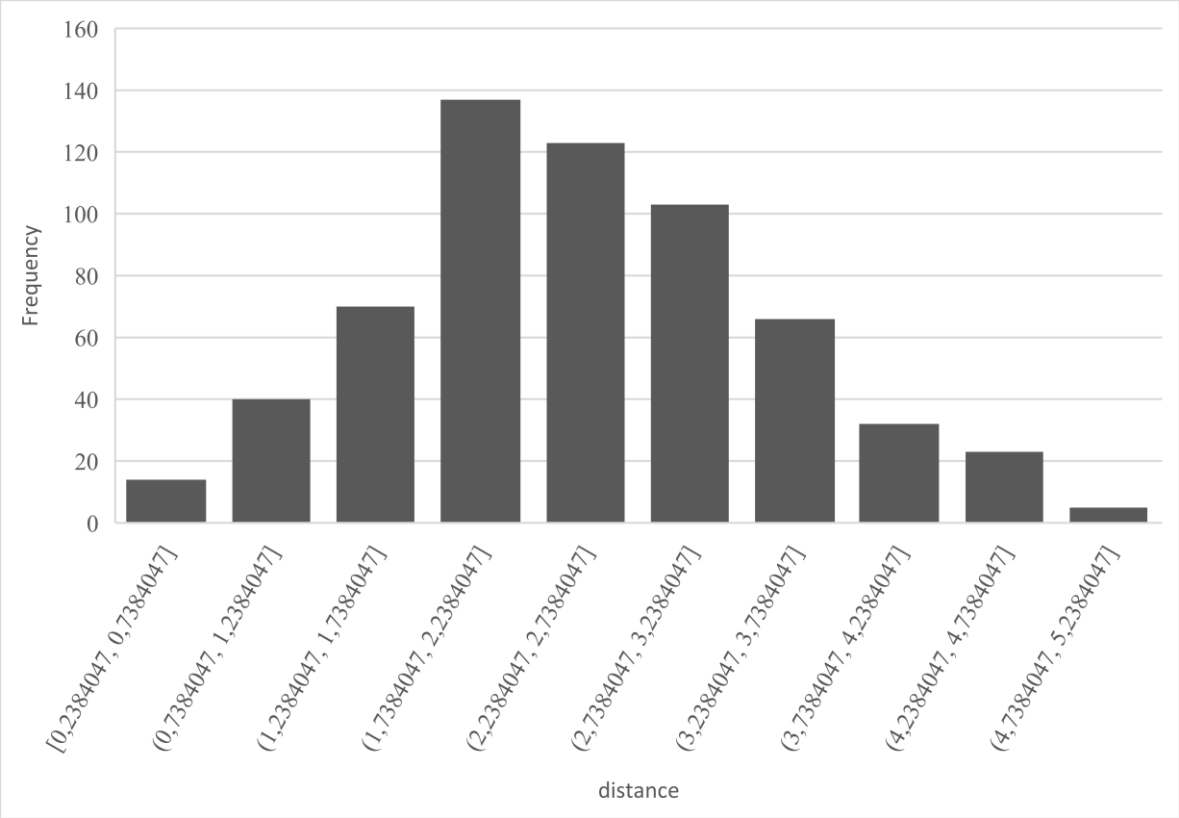

Figure 22: Frequency of distances from center point for edges of m_2 with l=2.04, r=0.1, m=613, and α=113.1°

To find out whether edges touch or are close to each other, how far they are from each other and on which side they are located, the Figures 23 to 30 are created. There, four distance values are checked for both models: 0.1, 0.05, 0.02 and 0.01. Only edges below these distances are counted. Figures 23 and 27 show that only very few edges are blocked from both sides. In Figures 26 and 30, most edges have only a small distance to nearby edges on both sides. Figures 23 to 30 again clearly show that the greater the distance from center, the lower the probability of blocking that edge. The comparison of m_1 with m_2 indicates a correlation of α-angle and m. The lower the α-angle, the fewer edges are needed to obtain unfolding resistance. This agrees with the observations from all experiments at this point.

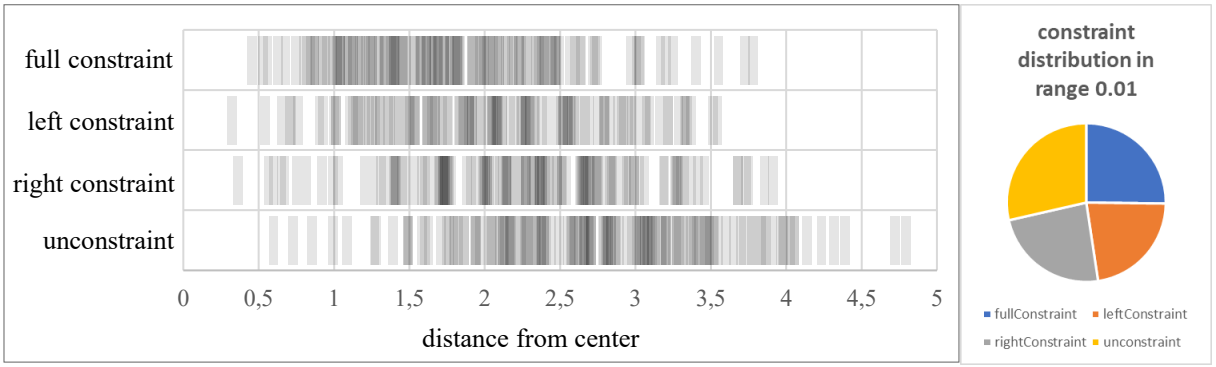

Figure 23: Constraints per distance from center in range 0.01 for m_1 with l=2.04, r=0.1, m=514, and α=93°

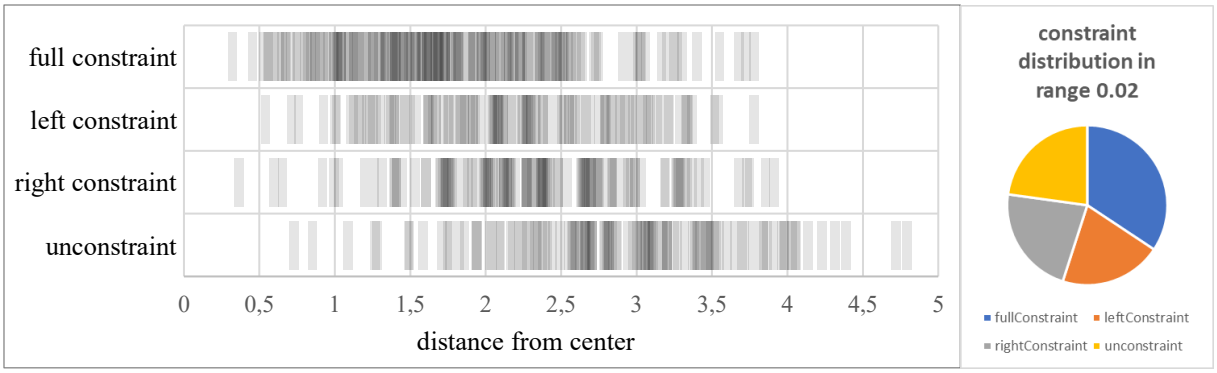

Figure 24: Constraints per distance from center in range 0.02 for m_1 with l=2.04, r=0.1, m=514, and α=93°

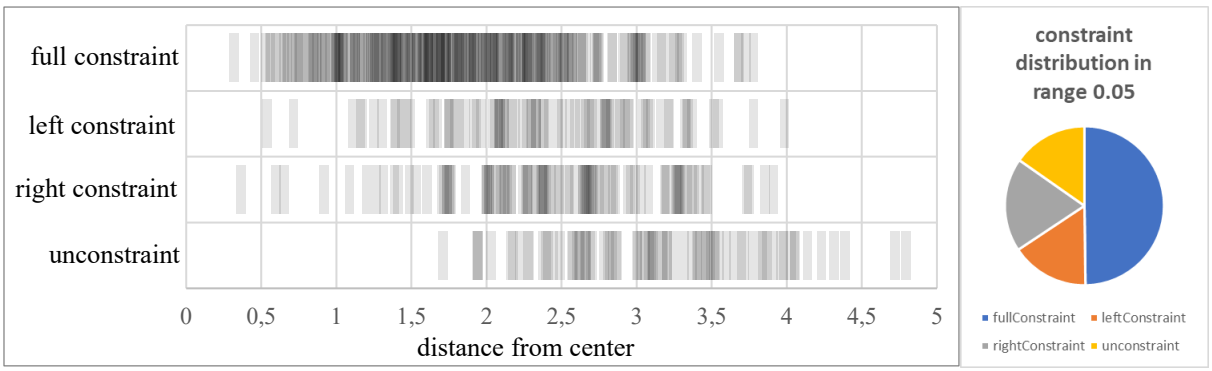

Figure 25: Constraints per distance from center in range 0.05 for m_1 with l=2.04, r=0.1, m=514, and α=93°

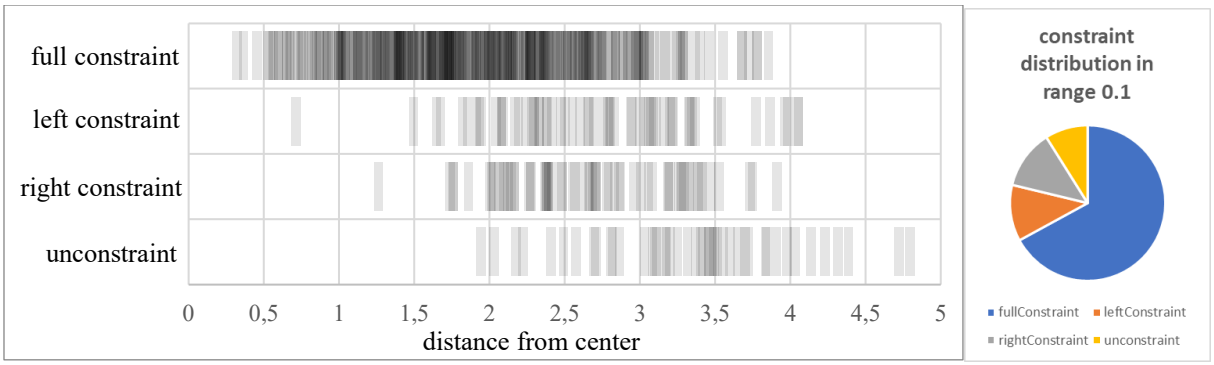

Figure 26: Constraints per distance from center in range 0.1 for m_1 with l=2.04, r=0.1, m=514, and α=93°

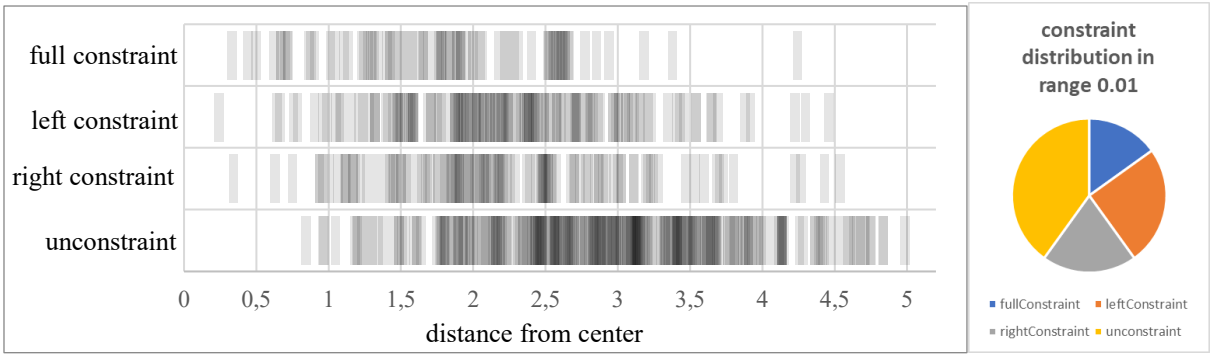

Figure 27: Constraints per distance from center in range 0.01 for m_2 with l=2.04, r=0.1, m=613, and α=123.1°

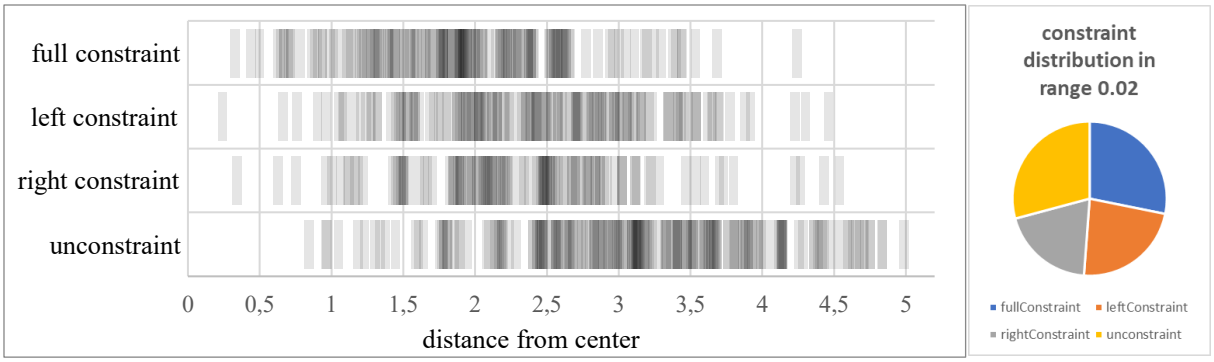

Figure 28: Constraints per distance from center in range 0.02 for m_2 with l=2.04, r=0.1, m=613, and α=123.1°

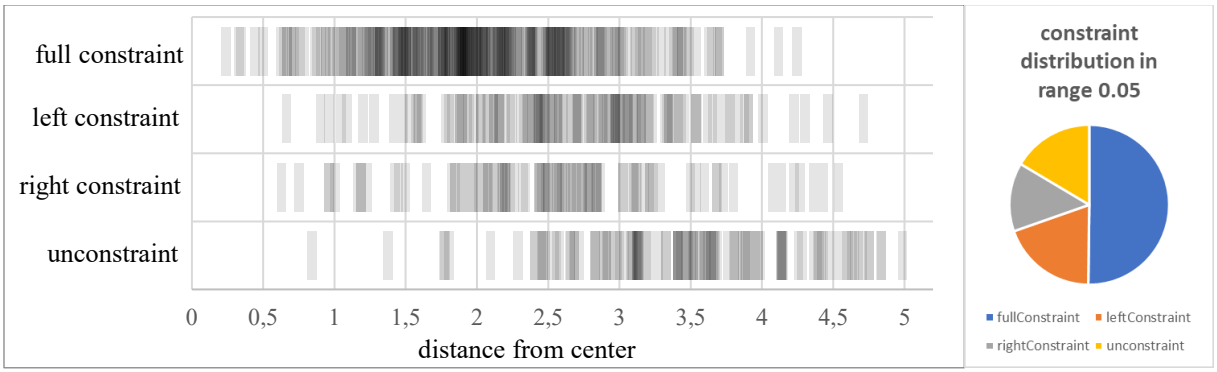

Figure 29: Constraints per distance from center in range 0.05 for m_2 with l=2.04, r=0.1, m=613, and α=123.1°

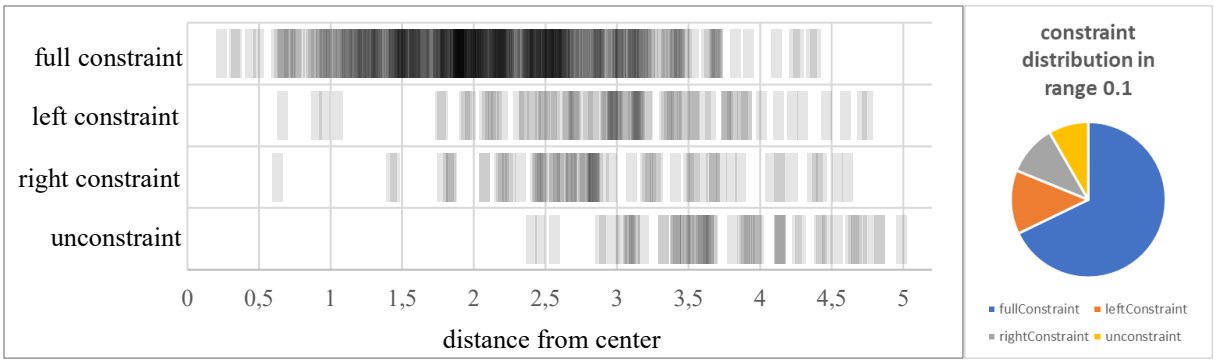

Figure 30: Constraints per distance from center in range 0.1 for m_2 with l=2.04, r=0.1, m=613, and α=123.1°

Further experiments with larger edge radius did not result in models that are locked or extremely unfolding resistant. Interesting observations were made anyway. Initially, the idea was that if the edge radius is increased, fewer edges at the same α-angle are needed to form unfolding resistance. However, it turned out that a certain complexity is needed for this to work at all. Perhaps there are too few interlocking angles of the chain because fewer edges can be placed in the sphere. Therefore, the α-angle must be increased, so that a similar construct can develop. It becomes clear that if such a construct with larger radius and α-angle is created, on the one hand the sphere must have a larger radius and on the other hand m must be much larger.

To get back to chains without radius Lemma 2 is used and the radius is set to an atomic ε to prevent intersections. From the previous observations, the following properties for large chains are concluded:

- The smaller α is, the smaller m can be, with the same unfolding resistance possible
- The smaller ε is, the smaller m can be, with the same unfolding resistance possible
- The higher the complexity inside and on the sphere (with most edges crossing its surface), the higher is the probability of a lock
- The smaller the distance to the surface of that sphere, the less movement in a dihedral joint is possible.

The idea of using physics simulation to create models has already been introduced. Because during the experiments it was noticed that the physics simulation can correct slight overlaps of edges by adjusting the dihedral angles, the idea from the chapter 6.1. At First Glance is considered again. It might be possible to generate a model with edges that partially overlap and partially do not touch, and then have the physics simulation make corrections. This leads to algorithm 5.

Algorithm 5: Corrected Intertwined Circular Path

Use algorithm 1

Transform the model into a physics model

Simulate the physics model with moving ground plate for 1 minute

If the model is stressless:

    Accept model

Else:

    Discard model

Therefore, the model from Figure 14 is generated again and then this is translated into a physics model. Corrections are made by the physics simulation, but it is found that there are tensions in the model, which can be clearly seen when a model starts to move by itself or if the boundaries of two half-edges that form an edge are visible in the dihedral joints in the physics simulation.

Then suitable parameters are found to create a testable model without stresses. Figure 31 shows this model. However, this model is prone to explosions in physics simulations because the beginning and end protrude from the model. If the model lands exclusively on the start or end edge, the joints will be overloaded. Therefore, the model has survived a maximum of two hours at a stretch in the physics simulation. No significant changes are detected. Locking is assumed after more than 50 runs without unfolding. In addition, it is found that m can be increased, and more edges can be sensibly placed, which also cannot come loose in many cases. The upper limit varies depending on the parameters used, because at some point edges are placed so that they overlap. $m = 24$ is used because this number of edges leads to a very unfolding resistant model.

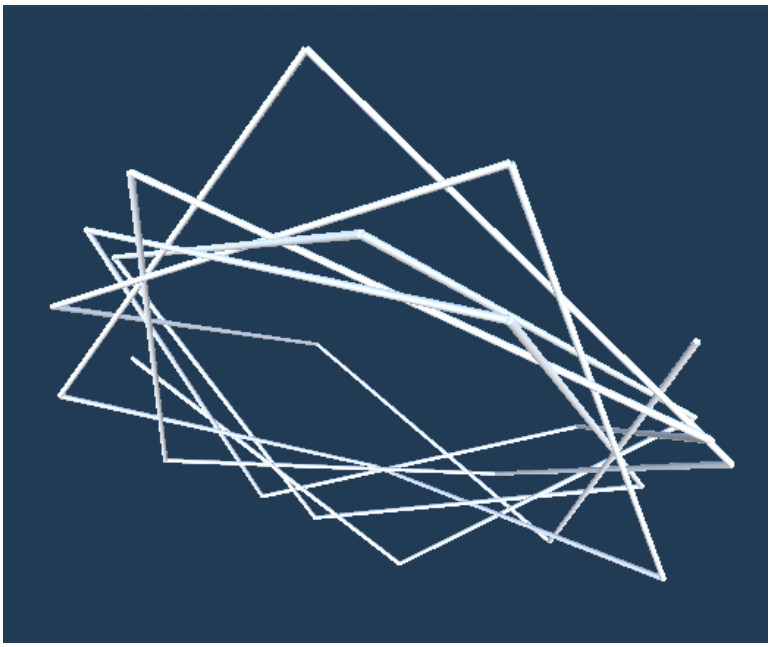

Figure 31: m_3 with m=24, l=2, r=0.025, α=91°, start-β=30°, and start-γ=3°

The exact number of edges does not play a special role in this case, because the edge radius is also important. If the edge radius is now reduced, all parameters must be adjusted to ensure locking. However, a special suitability of algorithm 5 is determined, which makes presumably locks with arbitrary edge radius possible. The search for suitable parameters turned out to be very time-consuming. Again, the larger the α-angle, the larger m should be.

So far, various ideas have been considered to prove locks. However, it is noticeable that these ideas have much in common. For example, edges are not placed on a spherical surface, but on a circle to reduce the number of edges. Algorithm 5 places all edges in the low-motion region, all with approximately the same distance to the central point. Most edges are placed so that these have at least one edge on both sides that is either blocked or very close to it. In particular, the beginning and end of the chain are placed sensibly. The chain does not have to be rigid, so slight movements of the chain are possible. Figures 32 to 38 show the analysis results for m_3. Figure 32 shows an increasing trend in contrast to Figure 17 and 18. Figure 33 shows that distances to the center point are uniform distributed over edge indices. Figure 34 shows that most midpoints of edges are placed near two circles c_1 and c_2. Where c_1 has a radius about $r_{c1} \approx 14.3$ and c_2 has a radius about $r_{c2} \approx 15.7$. In the middle region between the two circles, less than half of the midpoints of edges are detected then near the circles. Figure 35 clearly shows that so few edges are in direct contact that stresses can be excluded. Figures 35 to 38 show a smaller proportion of full constraints than in Figures 23 to 30 for all ranges.

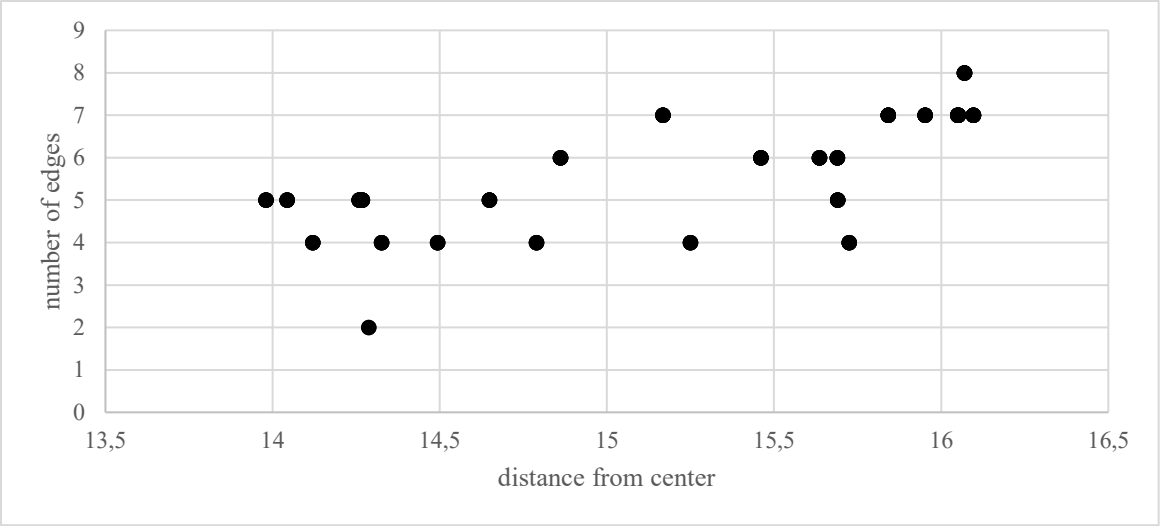

Figure 32: number of edges in distance one or less per distance from center for each edge of m_3 with l=2, r=0.025, m=24, α=91°, start-β=30°, and start-γ=3°

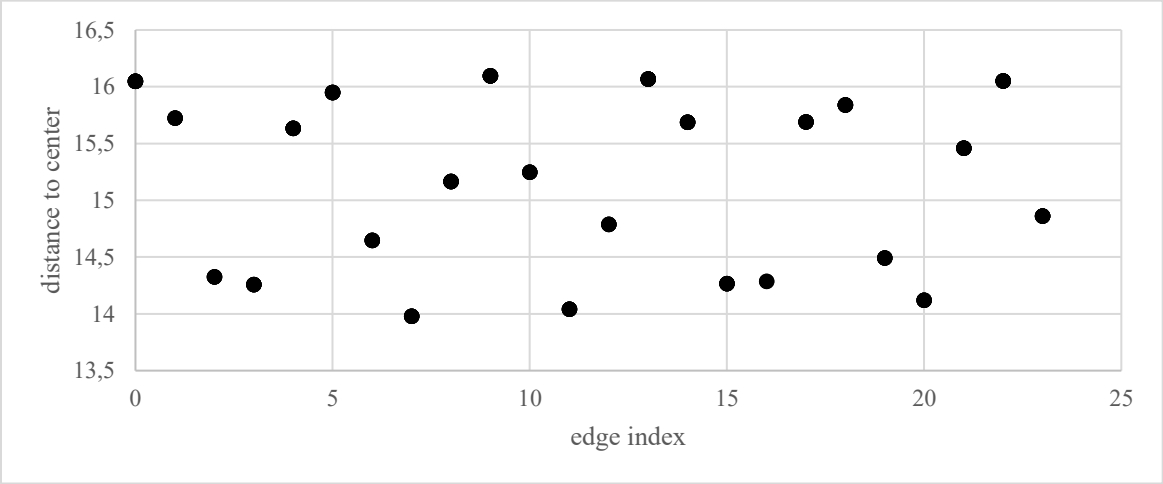

Figure 33: Distance to center per edge index for m_3 with l=2, r=0.025, m=24, α=91°, start-β=30°, and start-γ=3°

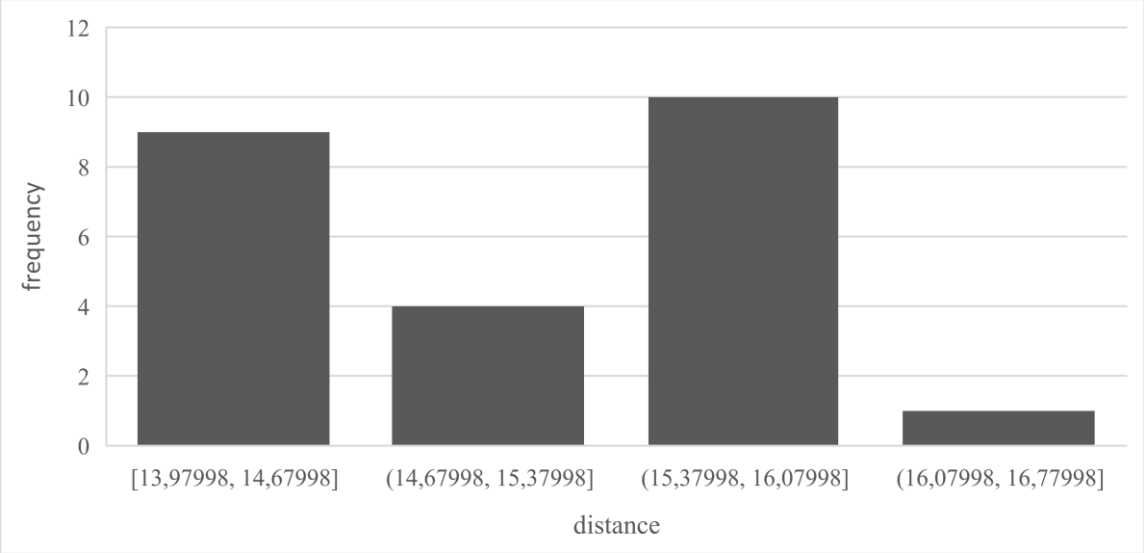

Figure 34: Frequency of distances from center point for edges of m_3 with l=2, r=0.025, m=24, α=91°, start-β=30°, and start-γ=3°

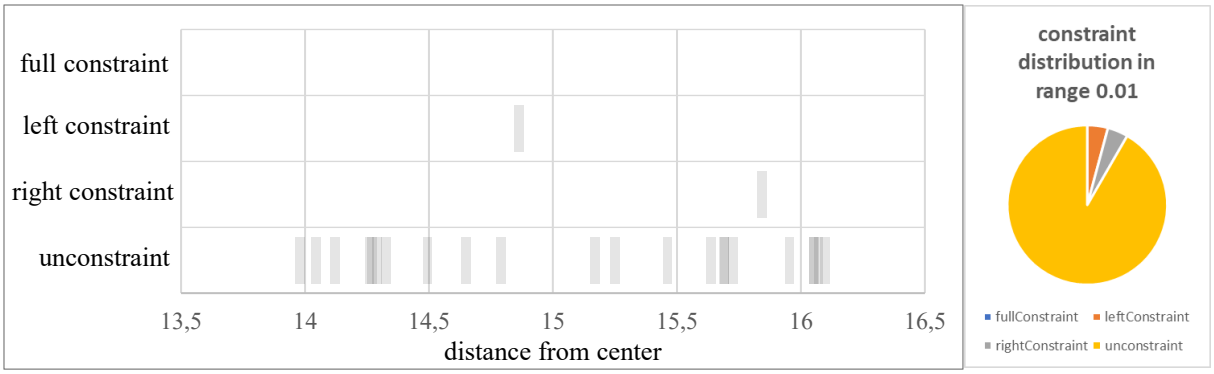

Figure 35: Constraints per distance from center in range 0.01 for m_3 with l=2, r=0.025, m=24, α=91°, start-β=30°, and start-γ=3°

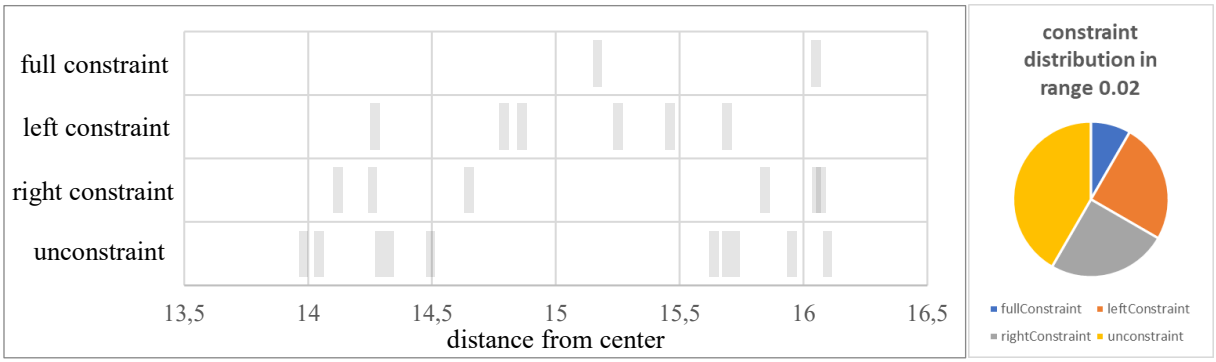

Figure 36: Constraints per distance from center in range 0.02 for m_3 with l=2, r=0.025, m=24, α=91°, start-β=30°, and start-γ=3°

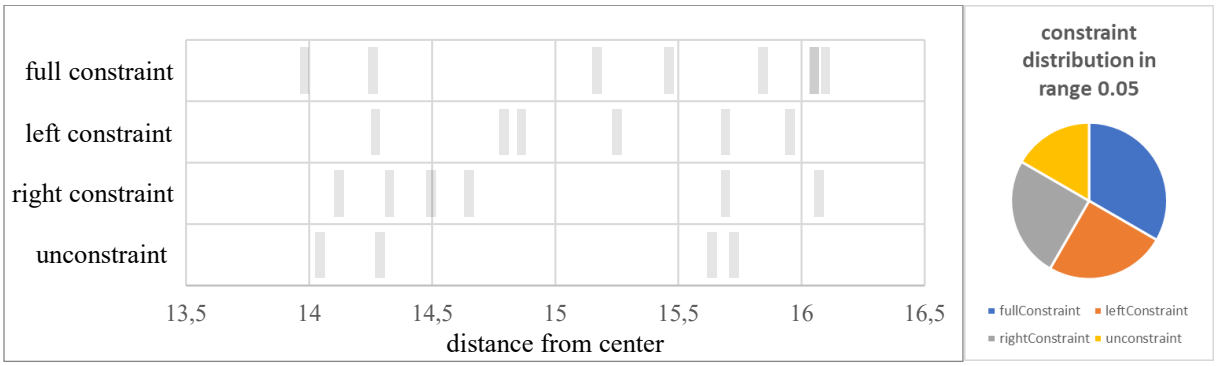

Figure 37: Constraints per distance from center in range 0.05 for m_3 with l=2, r=0.025, m=24, α=91°, start-β=30°, and start-γ=3°

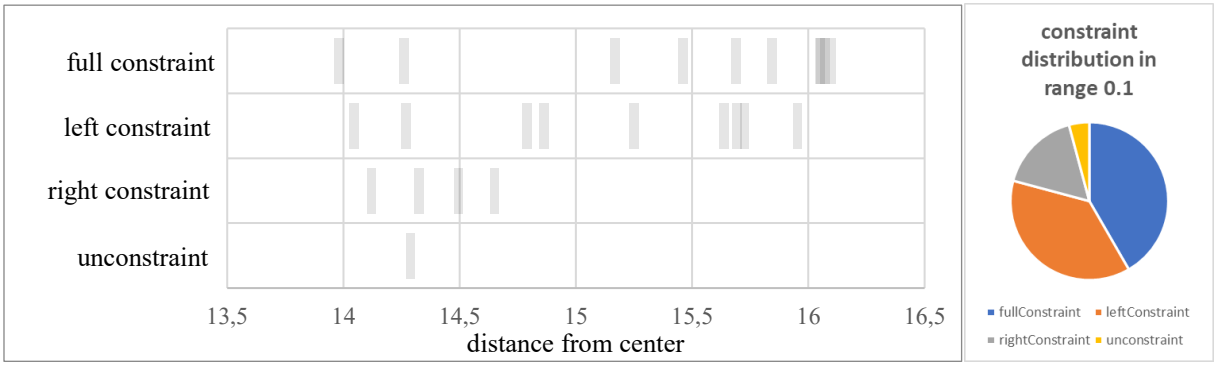

Figure 38: Constraints per distance from center in range 0.1 for m_3 with l=2, r=0.025, m=24, α=91°, start-β=30°, and start-γ=3°

Chains without a radius cannot be tested and the unfolding test is time-consuming. In addition, the test can only determine unambiguously if a model can be unfolded. Whether a model cannot be unfolded cannot be determined unambiguously with the test. However, time in the physics simulation that a model survives without unfolding can be considered as unfolding resistance. Therefore, no clear evidence is provided until now that locked open chains exist. However, such unfolding resistant models were found that a lock is assumed for them.

Even if this is not clearly proved, the assumption arises that if $n \to \infty$, then for all α-angles, there is a lock. The reason lies in the complexity. So, it seems logical that if n is larger, by the used algorithms, a stronger unfolding resistance can be reached, which probably ends in a lock. This results in Conjecture 7 and Open Problem 6.

Conjecture 7:

The probability that an open chain is locked increases with n. For $n \to \infty$ all open chains are locked.

Open Problem 6:

Prove or disprove Conjecture 7.

Over all experiments, non-flattenable configurations have turned out to be extremely rare for open chains with small n, randomly picked α-angle and Φ. So rare that no lock has yet been clearly identified, even if algorithms are used which search in promising configurations, instead of random sampling.

6.4. Simple Locks

While playing with a wooden model in a trefoil knot configuration with minimum number of edges and thinking about the knitting needles example given by Erik Demaine and Joseph O'Rourke in the context of locked open chains without fixed angle, obtuse, equiangular, and equilateral restriction, [44] a final idea emerged. Three edges were removed from the wooden model, and the α-angle got slightly reduced, and it still seemed very resistant to unfolding. Figure 39 shows the knitting needles idea and Figure 40 shows the new wooden model. In the following, locks based on this idea are referred to as simple locks. A simple lock configuration is given, if the corresponding model is non-flattenable and looks like the wooden model in Figure 40.

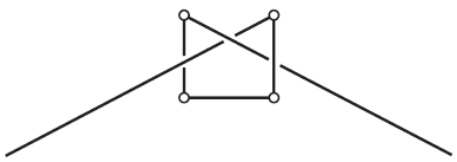

Figure 39: Knitting needles

Source: Figure 6.2 in Geometric Folding Algorithms: Linkages, Origami, Polyhedra [44]

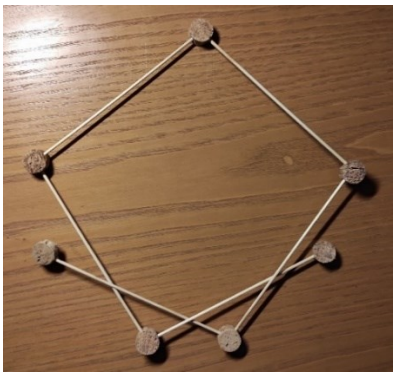

Figure 40: Wooden model with m=6, l=19, r=0.15 and α≈92°.

All physics models considered now have $l = 2\ and\ r = 0.025$. Because, previously, it was found that low α-angles require fewer edges for locks, the α-angles 90.5° and 92° for $m = 6$ were checked first with physics models. It turned out that these physics models can unfold. It was noticed that for unfolding, the key movement is a bending up of the top node of Figure 40. Then various α-angles were tested in range 90° to 108°. This range is selected because there start, and end edge can cross based on the formula for regular polygons. It turned out that in this configuration, an increase in the α-angles resulted in a decrease in freedom of movement. Somewhere between 92° and 92.5° the border between unfolding and locked models is detected. The border between locked and stressed models is between 104° and 104.5°. A locked model for $m = 6$ is shown in figure 41. Then all $m \leq 10$ were tested. The results are shown in Table 2. Based on Lemma 2, if the radius of the edges decreases, the freedom of movement of the model increases. It follows that a model is locked in a smaller α-range when the radius decreases. At the same time, the α-range in which a model is stressed becomes smaller and the α-range in which a model is flattenable grows. Therefore, it follows that any chain with

$$\alpha_{simple\ lock\ max}(m) = \pi \cdot \frac{m-3}{m-1} - \varepsilon \qquad (10)$$

has a rigid configuration or a configuration in a very small configuration subspace and is thereby locked. In addition, there are some more α-angles with

$$\alpha_{simple\ lock}(m) < \pi \cdot \frac{m-3}{m-1} \qquad (11)$$

and low difference to $\alpha_{simple\ lock\ max}$ that are locked. The higher m is the lower is the additional locked range. Every chain can be locked at possible lock angles of all smaller chains. For example, for $m = 7$ models are locked within $92.5° \leq \alpha \leq 104°$ and $115.5° \leq \alpha \leq 116.5°$.

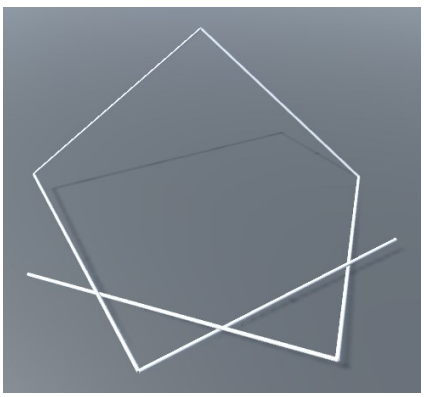

Figure 41: Smallest locked open physics model with l=2, r=0.025, m=6, α=94° and $\Phi_{start} = (-45°, 45°, -34°, 12°)$

In Figure 42, the α-range in which the start and end edges can intersect in the largest possible circle is compared to the α-range contained within where a model is locked. Especially the jump between $m = 6$ and $m = 7$ is striking, where the ratio drops from more than $\frac{3}{4}$ to $\frac{1}{6}$. It is concluded that for $m = 6$ there is an increased restriction of freedom of movement in the simple lock configuration. Whether the ratio remains the same for all $m > 6$ or slowly decreases cannot be clearly determined now. For this, more precision in the locked range and a wider range of edge numbers must be tested.

Table 2: Observations for physics models based on simple locks with l=2 and r=0.025

range: 90 to 108			range: 108 to 120			range: 120 to 128,57			range: 128,57 to 135			range: 135 to 140		
m	**α**	**observation**	**m**	α	**observation**	**m**	α	**observation**	**m**	α	**observation**	**m**	α	**observation**
6	106	stressed	7	118	stressed	8	127	stressed	9	134	stressed	10	138.5	stressed
6	105	stressed	7	117	stressed	8	126.5	stressed	9	133.5	stressed	10	138	locked
6	104.5	stressed	7	116.5	locked	8	126	locked	9	133	locked	10	137.5	unfolds
6	104	locked	7	116	locked	8	125.5	locked	9	132.5	locked			
6	100	locked	7	115.5	locked	8	125	unfolds	9	132	unfolds			
6	96	locked	7	115	unfolds	8	124	unfolds						
6	94	locked	7	114	unfolds	8	122	unfolds						
6	93	locked												
6	92.5	locked												
6	92	unfolds												
6	90.5	unfolds												

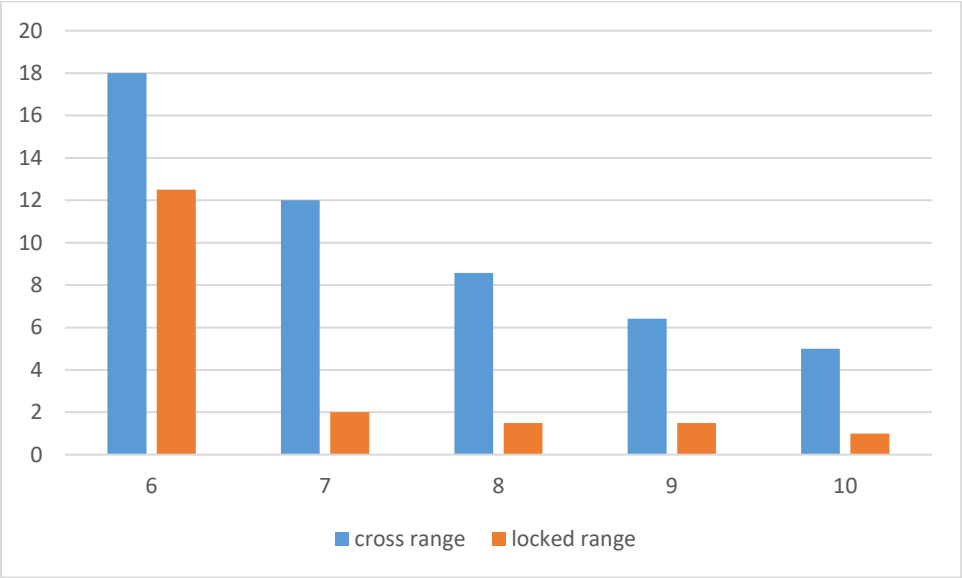

Figure 42: Crossing α-range for first and last edge based on the regular polygon equation and locked α-range rounded up with 0.5° accuracy at the borders for physics models based on knitting needles with l=2 and r=0.025

Models with low edge radius to edge length ratios such as the model in Figure 41 could be created in a non-flattenable configuration. Based on Table 2, it is found that there is a progression from impossible or stressed configurations through locked configurations to unfolding or flattenable configurations based on α-angles in the range $\pi \cdot \frac{m-4}{m-2} < \alpha < \pi \cdot \frac{m-3}{m-1}$. Larger α-angles are stressed, and smaller α-angles are flattenable in these configurations. If the radius to length ratio of the edges is reduced, it follows based on Lemma 2 that the α-range in which a model is stressed becomes smaller and at the same time the α-range in which models in such configurations are flattenable grows. Another point is

that not only $\alpha_{simple\ lock\ max}(m)$ is locked, but it is always followed by an α-range in which a model is not rigid but is still non-flattenable. If the radius to length ratio is now reduced to an infinitesimal $\varepsilon_{r/l}$, it follows that the stressed α-range gets very small. ε_α is the range of stressed configurations and is based on $\varepsilon_{r/l}$. The smaller $\varepsilon_{r/l}$, the smaller ε_α, where if $\varepsilon_{r/l} \to 0$, then $\varepsilon_\alpha \to 0°$ as well. Additionally, if a non-flattenable chain is found and it is extended by one edge, then the resulting chain must also be non-flattenable. Smallest simple locks specify the minimum number of edges for chains that have certain α-angles. If a chain has less edges, it is excluded that it is locked. If any smallest simple lock is taken, and an edge is removed, then only start and end edges can touch. For a lock in this configuration the start and end edge must be touched by at least two edges each. Thus, any separation of the configuration space is clearly excluded. Conjecture 8 and Open Problem 7 summarizes this.

Conjecture 8:

Any open chain with $\alpha_{simple\ lock\ max}(m) = \pi \cdot \frac{m-3}{m-1} - \varepsilon$ has a rigid configuration or a configuration in a very small configuration subspace and is thereby locked. In addition, there is a small range with $\alpha_{simple\ lock}(m) \leq \alpha_{simple\ lock\ max}(m)$ that is locked based on lack of freedom of movement in non-flattenable configurations. All locks that exist with smaller m are also valid for larger m with the same α-angles. For the resulting α-angles, all open chains with fewer edges than m are not locked.

Open Problem 7:

Prove or disprove Conjecture 8.

6.5. Configuration Space

Using a 3D printer, a model was created that is more accurate than wooden models. If this is taken in simple lock configuration and tried to flatten it by hand, one again quickly notices that this is not possible. Another confirmation is that the configuration subspace of a simple lock is relatively small. Even if scanning of the entire configuration space is intractable, we will try to scan the local configuration subspace. If a wrapping of the local configuration subspace by self-intersections is detected, a lock under the sampling accuracy is proved. This is not a general proof, because it does not prove that there is no configuration without self-intersection that is not covered by the grid. In addition, relatively thick edges must be used, because otherwise it cannot be guaranteed that self-intersections cannot be skipped. However, this is already a good hint, and we can have a look at the configuration subspace.

Fortunately, a lock with only four dihedral joints is found. Therefore, its configuration space can be represented as a 4D cube, which represents a torus. Every coordinate in this cube represents a configuration and for every point it is indicated if this has a self-intersection or not. We can think of this 4D cube essentially as a 3D black and white film scene that changes over time. We call a single frame of this 3D movie a slice. Each dihedral joint can take on an angle between 0° and 360°, which is why the entire configuration space corresponds to a cube with $(360°/stepsize)^4$ points or that we have a 3D scene of size $(360°/stepsize)^3$ and the whole movie consisting of $360°/stepsize$ slices. If only the configurations that do not lead to a self-intersection are considered in this 4D cube, there are two possibilities. Either all components of this 4D space are connected under the torus property of the space, or not. To understand this, let us now imagine a single slice. If this slice contains, for example,

a filled sphere, which is surrounded only by configurations with self-intersections, and there are other configurations without self-intersections that are not in the sphere, we know that if one joint corresponding to the removed dimension has a fixed angle, the chain is locked. If we now look at the entire 3D film and follow our sphere, and the sphere in each slice represents a separate component, whereby this must always overlap, but must not overlap with non-self-intersecting points outside the sphere, in any two consecutive slices, we know that there is a lock. In the optimal case a lock can be determined, in which all slices are overlapped. If the result contains again a split off sphere, a lock of a chain can be proved under $stepsize$.

This procedure is not only applicable to 4D spaces, but also to any other. We can imagine a sphere in 3D as a normal black and white film. The film consists of 2D photography's. The sphere is represented by filled circles or a single point in a 2D photography. 2D circles can be represented by parallel lines in 1D. Essentially, a space is decomposed by reducing its dimension by one and considering the removed spatial dimension as a time axis. The procedure can be used recursively to decompose arbitrary dimensional spaces into 1D spaces and to answer questions there. If it is then true that all composite 1D spaces containing the isolated configuration subspace, represent 2D filled circles or a point, these belong to 3D spheres and the 3D spheres are representing a 4D body with an isolated configuration subspace. It is important that the component does not have to be a sphere, but just an arbitrary entity that is clearly separated from other configurations without self-intersections. If this entity is convex the 1D checks are simpler. This roughly describes the idea behind CAD where cylinders in polynomials are used, but we will use simpler algorithms.

We use only locale dihedral motions to move around the cube of the configuration space. This automatically creates a grid. Each coordinate $(j1,\ j2,\ j3,\ j4)$ of the grid has eight neighbors. A neighbor can be reached by walking one step along an axis in either a positive or negative direction. So, the neighbors are $j1_+$, $j1_-$, $j2_+$, $j2_-$, $j3_+$, $j3_-$, $j4_+$, and $j4_-$. The distance to each neighbor is $stepsize$. This is an example for 4D, but this works in arbitrary dimensions. There the number of neighbors equals two times the dimension. [40]

The first idea is to scan a small area in the form of a 4D cube, with the start configuration in the center of this cube. The start configuration (-45°, 45°, -34.2°, 17°) is used and a cube with an edge length of about 4° is scanned for self-intersections with 0.1° steps. A model with $m = 6$, $\alpha = 94°$, $l = 2$ and $r = 0.05$ is used. Unfortunately, during the sampling it was noticed that either $stepsize$ is selected too large or that the l to r ratio must be adjusted, because in a very rare case self-intersections can be skipped. The results of the scan are analyzed anyway. It is first determined that plotting large point clouds and moving in or around them is not so easy with Unity, nor with Matplotlib. Therefore, the configuration space is mainly studied with PyQtGraph, which is an OpenGL based python library. In tests 13 million data points could be displayed and the camera control does not freeze. To view the 4D space, the first three joints of all 4D coordinates are displayed. Coordinates with self-intersection are marked green and coordinates without self-intersection are marked blue. If there are configurations with and without self-intersection on the same coordinate, it is marked teal. If several data points are superimposed from the current perspective, the overlay of green and blue forms teal. If a point that is teal is overlaid with green or blue points, teal always dominates. The first key observation is that even when all scanned 3D slices are overlaid, the configuration space clearly has regions containing only self-intersections that separate other regions containing both self-intersections and non-self-intersections. However, it must be considered that not all slices, but only the 4° were overlaid, so it is not clear whether this applies when 360° are overlaid. In addition, it is found that the configuration space in this form is cut by plates with self-intersections. The start configuration is in an area which is bordered by at least two such surfaces and additionally borders on a large area with exclusive self-intersections. Figure 43 shows the chain in start configuration. Figure 44 illustrates three perspectives showing the surrounding borders in the 4° configuration space around the start configuration, where the start configuration is white or red. In the first image, there is a large green area adjacent to the start configuration. The green area can be described as a part of an area that lies between the surfaces of

two spheres with different radii, like a slightly curved plate. In the second image, there is a 0.2° thick green plate to the left of the start configuration, and in the third image, there is a 0.3° thick green plate to the right of the start configuration. In addition, in the second and third images, three green 0.1° plates are visible to the right of the start configuration and to the left of the 0.3° plate.

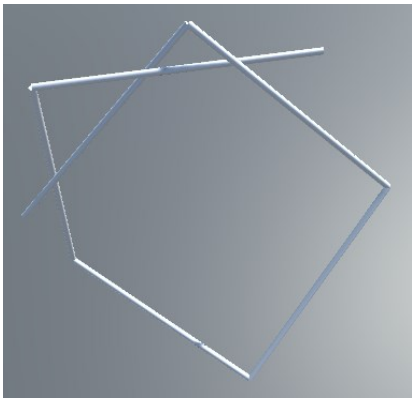

Figure 43: Open model with m=6, α=94°, l=2, r=0.05 and Φ = (-45°, 45°, -34.2°, 17°).

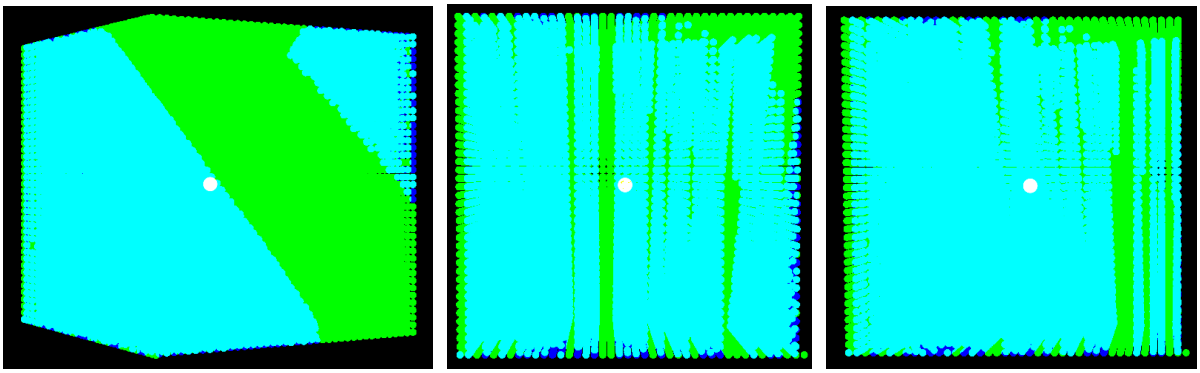

Figure 44: Three perspectives of the 4° hypercube overlay in 3D using (j1, j2, j3) as coordinates with $stepsize$=0.1° of a chain with m=6, α=94°, l=2, r=0.05 and Φ_{start}= (-45°, 45°, -34.2°, 17°), where boarders of the local configuration subspace are shown.

Because the sampling of a cube section is unsuitable for the expected shape of the local configuration subspace, the configuration space is now sampled differently. We search only the coordinates that belong to the local configuration subspace or those that are adjacent to it. If we find a contiguous hull, we are even guaranteed that there is no 4D escape from the overlay, which could not be guaranteed before.

To investigate the space around the start coordinates, we will use breadth-first search. Therefore, we will need two lists. The first list we call $ToTest$ and the second $Tested$. In $ToTest$ we store indices of coordinates in Tested, which we have found without self-intersection. In $Tested$ we store all coordinates that we have already examined. We think of $ToTest$ as a list of all coordinates whose neighbors still need to be tested. A simple lock configuration is selected as start configuration. Therefore, this is added to $ToTest$ and $Tested$ after verifying that it does not lead to self-intersections. In each step the first index is removed from $ToTest$ and the corresponding coordinate is found in $Tested$. Then the neighbors of this coordinate are examined, that are not contained in $Tested$, and added to $Tested$. The indices of the tested coordinates without self-intersections are then added to $ToTest$. The algorithm terminates in each case, either after the entire configuration space has been scanned, or after isolation of the locale configuration subspace has been detected. We export the coordinates with self-intersections to one list and the coordinates without self-intersections to another list. Algorithm 5 describes this for 4D, however theoretically arbitrary dimensional configuration spaces can be examined for connectedness under $stepsize$, if the number of neighbors is adjusted.

Algorithm 5: Configuration Space Connectivity Test under local dihedral motion with step size in 4D

Input: Φ_{start}, $stepsize$

Output: local configuration subspace or entire configuration space

Create the empty lists $ToTest$<int> and $Tested$<float[4]>

Create a model with Φ_{start}

Test the model on self-intersections

If no self-intersections are detected:

    Export Φ_{start} based on self-intersections

    Add Φ_{start} to $Tested$

    Add its index to $ToTest$

Else:

    Exit (no valid start configuration)

While $ToTest$ is not empty:

    Get and remove the first element from $ToTest$

    Use it to get the corresponding coordinates Φ_{ToTest} from $Tested$

    Create new configurations Φ_1 to Φ_8 corresponding to the eight neighbors of Φ_{ToTest}

    Foreach Φ_i of the eight neighbor configurations:

        If Φ_i is not in $Tested$:

            Create a model with Φ_i

            Check the model on self-intersections

            Export Φ_i based on self-intersections

            Add Φ_i to $Tested$

            If no self-intersections are detected:

                Add index of Φ_i in $Tested$ to $ToTest$

A new model, m_4, is created in simple lock configuration, with $m = 6$, $\alpha = 97°$, $l = 2$, $r = 0.08$ and $\Phi = (-22°, -2°, 20°, -45°)$, which is shown in Figure 45. Then Algorithm 5 is used with this model. It is found that the local configuration subspace is larger than expected after all. In addition, it is also observed after nine days of computation that $stepsize$ is too large and a self-intersection could be skipped. Therefore, the calculation is then stopped. A total of 1.432.574 4D coordinates were exported, of which 1.121.108 coordinates have no self-intersection and 311.466 coordinates belong to self-intersections. Nevertheless, the results provide an interesting insight into the configuration space.

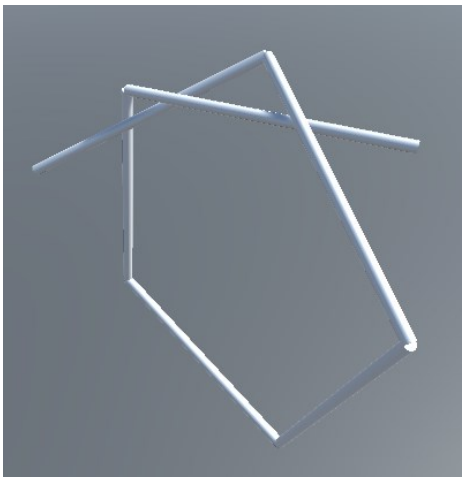

Figure 45: Open model m_4 with m=6, α=97°, l=2, r=0.08 and Φ = (-22°, -2°, 20°, -45°).

Because the algorithm did not terminate and in late samples few points without self-intersection were determined, which were reached by skipping a self-intersection, we must first check how this affects the determined data. The points that were wrongly sampled are in an area that is far away from the start coordinate. Probably on the surface of the detected 4D area. These points are likely to be on the boundary of the found hull or on the surface which should still be explored. This is because we are using broad search. The slices that are closer to the start coordinate are larger than those that are further away. Figure 46 shows three slices where $(j1, j2, j3)$ are used as coordinates and the slices $j4 = -45°$, $j4 = -47.5°$, and $j4 = -50°$ are shown.

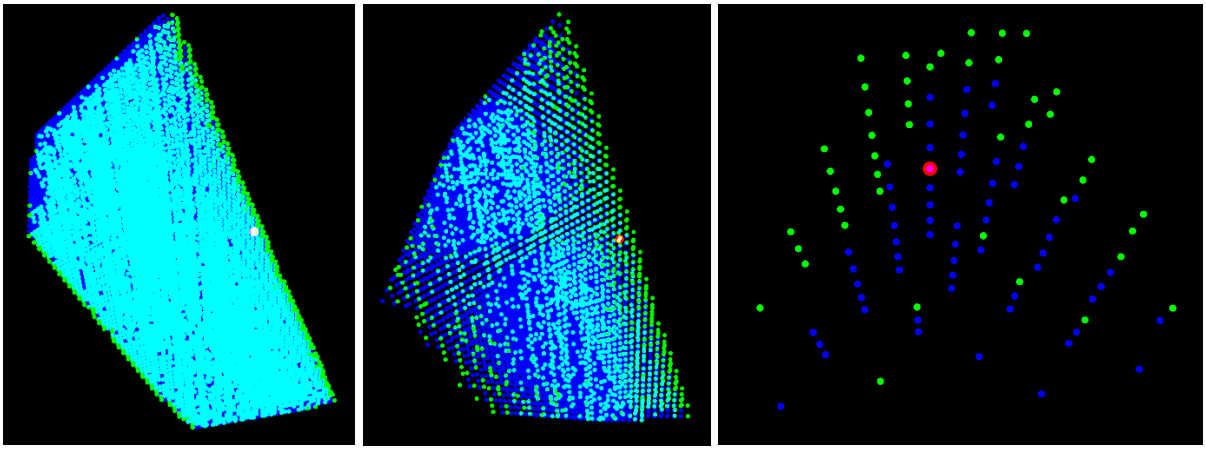

Figure 46: Slices j4=-45°, j4=-47.5° and j4=-50° of the configuration space results of m_4.

A particularly interesting observation is that the sampled area is convex in all slices. In the larger slices, three boundary surfaces are in most cases visible, which belong to the hull of coordinates with self-intersections. These surfaces have the shape of planar convex polygons. This is true for all investigated 3D slices, which can be formed by any combination of $j1$, $j2$, $j3$, and $j4$. Surprisingly, the same is true for the overlays of all slices. Figure 47 shows the overlay of all $j4$ slices, which look like the $j4 = -45°$ slice. Figure 48 shows four overlays of all slices where always another joint is used as time axis. Only coordinates without self-intersection are displayed in that case. When the observations are combined, Conjectures 9 and 10, and Open Problems 8 and 9 are obtained. If Conjecture 9 is true, then it follows that the local configuration subspace of simple locks is convex. Furthermore, it should be easy to represent it with equalities and inequalities because it has a simple surface structure.

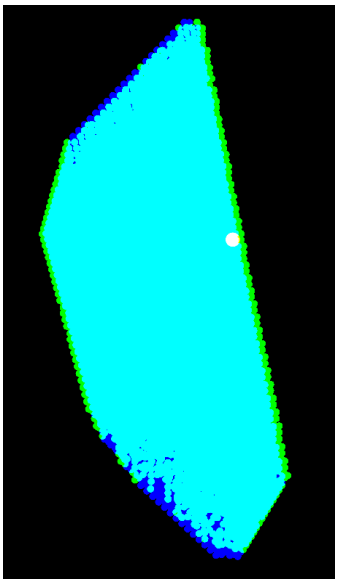

Figure 47: Overlay of all j4 slices of the configuration space results of m_4.

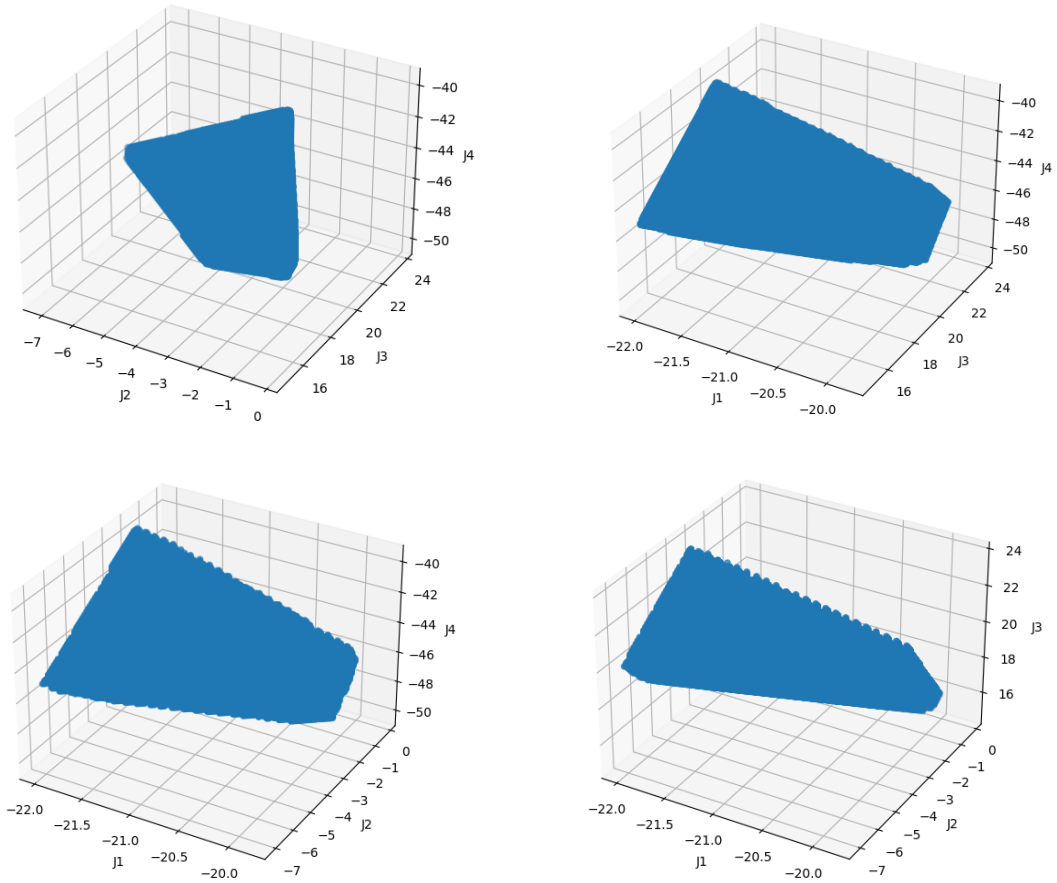

Figure 48: Overlay of all j1, j2, j3, and j4 slices where only coordinates without self-intersection are displayed of the configuration space results of m_4.

Conjecture 9:

Any slice of any dimension of the local configuration subspace of a simple lock has a convex hull.

Open Problem 8:

Prove or disprove Conjecture 9.

Conjecture 10:

Any overlay of slices of any dimension of the local configuration subspace of a simple lock has a convex hull.

Open Problem 9:

Prove or disprove Conjecture 10.

$stepsize = 0.1°$ and $\alpha = 97°$ seems to be reasonable, so it is recommended to slightly increase the radius to length ratio of edges for further investigations. This will make the local configuration subspace slightly smaller. In addition, an algorithm should be used that only scans the surface of self-intersections, because the configuration subspace of simple locks has a larger volume than initially expected.

6.6. Producible Protein Chains

Question 6:

Are there classes of open chains that cannot be produced by a ribosome? Answered by Erik Demaine, however, here again considered for found locked chains. [13]

To give a brief introduction to the subject of protein chains, Erik Demaine is cited:

"… fixed-angle polygonal chains in 3D may serve as a model of protein backbones, with each vertex an atom and each link a covalent bond between atoms, with nearly fixed bond angles. We detail this model a bit more before proceeding. Each amino acid residue in a protein backbone consists of three atoms: nitrogen followed by two carbons, conventionally named $N - C\alpha - C'$. As the name implies, the backbone is only the central core: H attaches to N, O attaches to C_α, and a 'side chain' (which differs for each amino acid) attaches to C_α. We will only model the backbone without attachments. The residues are joined by peptide bonds, and the full structure is often called a polypeptide. All the bond lengths along the backbone (link lengths in the model) are approximately fixed, at values between 1.33 and 1.52 Å, that is, roughly the same length. The bond angles are again approximately fixed angles:

$$\angle NC_\alpha C \approx 109.5°, \angle C_\alpha CN \approx 116°, \angle CNC_\alpha \approx 122°.$$

Again, note that it is not too far wrong to treat all bond angles as roughly equal. In our polygonal chain model, we permit free spinning (dihedral motions) about each link, but in fact the $C' - N$ bond only permits two values: 0° and 180°, resulting in two planar configurations (known as trans and cis). Thus, the chain is 'partially rigid' in our earlier notation. However, we will ignore this partial rigidity,

modeling protein backbones as fixed-angle polygonal chains, and when convenient, with equal joint angles and equal link lengths." [5]

To check which configuration subspaces of locked open chains found can be produced by a ribosome, Theorem 2 by Erik Demaine et. al. is used. Theorems 2 and 3 are based on a simplified ribosome model, where the chain is produced inside a cone at its tip. All attempts to classify locks presented so far are based on configurations that are assumed to be non-flattenable. Based on Theorem 1 all these configurations are not producible if these are non-flattenable.

By Theorem 3 we know that the larger n of an open chain is, the more likely it is that it is non-flattenable when it is locked. Combining our observations with Theorem 3 leads to the assumption that the configuration space of chains with simple lock α-angles for very large n consists mainly of non-flattenable configurations and that this is true for all chains, where n must be relatively larger to achieve the same percentage of non-flattenable to flattenable configurations as for simple locks.

The given angles are not clearly in the simple lock category. It might be that $\alpha_{average} \approx 115.8°$ has simple locks. In addition, it is not clear, if simple locks really exist for chemical bonds which are not exactly equiangular and equilateral in most cases.

Theorem 2 (Erik Demaine et. al.):

A configuration of an open chain is producible if and only if it is flattenable.

[13]

Theorem 3 (Erik Demaine et. al.):

If there is a chain, that is locked, the probability that a random configuration of that chain is non-flattenable approaches one if $n \to \infty$.

[13]

6.7. Cutting Chains

Question 7:

How many cuts are necessary to unlock a chain or all resulting chains? This question is an adaption of a question from Anna Lubiw. [14]

Interlocks are not considered in the partial answer to Question 7. However, these must be considered to fully answer it. Based on what we have found, when a locked closed chain is cut, this does not necessarily mean that the resulting open chain is not locked. The same is true for locked open chains. For example, a simple lock could be used where the number of edges is doubled. If this chain is cut in the middle, two open chains are created, which are both locked. Based on Conjecture 8, if the number of edges is too small for a simple lock, then this chain is not locked. So, cutting an open chain into parts that have too few edges for a simple lock, guarantees that all resulting chains are not locked. This results in Conjecture 11. If Conjecture 8 is proved, Conjecture 11 becomes a corollary.

[5] E. Demaine and J. O´Rourke, "Poducible Polygonal Protein Chains", in *Geometric Folding Algorithms: Linkages, Origami, Polyhedra*, Cambridge University Press, 2007, p. 148. [5]

Conjecture 11:

For all chains which have an α-angle of a simple lock, at least so many cuts are required that each resulting chain has fewer edges than are needed for a simple lock to ensure that all resulting chain parts are not locked. All chain parts are examined in separated rooms, which does not take interlocks into account.

6.8. Interlocked Open Chains

Interlocks are locks that are created by several chains that are not isolated and their interaction, where this is interesting if each chain is not locked by itself. Only Open Problem 10 is formulated for this topic based on parallel production of protein chains. Multiple robot arms in the same environment can also benefit from solving this problem.

Open Problem 10:

Do two open chains exist, which are not locked by themselves and have same α-angles and edge lengths, but can be locked by interactions? If two chains are not enough, how many open chains are at least necessary.

6.9. Open Chains Summary

Open chains can be divided into two groups:

$$\frac{\pi}{2} < \alpha_{maybe\ locked}(m) < \frac{\pi \cdot (m-3)}{m-1} \leq \alpha_{not\ locked}(m) < \pi \tag{12}$$

Open chains with $\alpha_{maybe\ locked}(m)$ can be divided into the two groups $\alpha_{simple\ lock}(m)$ and $\alpha_{not\ simple\ lock}(m)$:

$$\frac{\pi \cdot (m-3)}{m-1} - x(m) < \alpha_{simple\ lock}(m) \leq \frac{\pi \cdot (m-3)}{m-1} - \varepsilon \tag{13}$$

$$\alpha_{maybe\ locked}(m) \setminus \alpha_{simple\ lock}(m) = \alpha_{not\ simple\ lock}(m) \tag{14}$$

$x(m) > \varepsilon$ is the range of simple locks, which is not precisely determined.

Chains with $\alpha_{not\ locked}(m) \neq \frac{\pi \cdot (m-3)}{m-1}$ are not locked, because a maximum of two edges can touch. Those chains are always flattenable. If $\alpha_{not\ locked}(m) = \frac{\pi \cdot (m-3)}{m-1}$ the chain cannot form a simple lock.

Chains with $\alpha_{simple\ lock}(m)$ are locked. This is based on observations in physics simulations, real world observations of wooden and 3D printed models and findings from scanning the configuration space of the smallest simple lock.

It is assumed that the probability that an open chain is locked increases with n. For $n \to \infty$, then all chains are locked.

It has been noticed that over all observations non-flattenable configurations are rare if n is small.

Representing the configuration space as a cube with torus property is very useful. It is assumed that the local configuration subspace of simple locks has a convex hull in this representation form. The borders of this hull consist of convex planar polygons, whereby possibly a part of these polygons is not planar but has a slight curvature.

Open chains are producible if these are flattenable. If there is a chain, that is locked, the probability that a random configuration of that chain is non-flattenable approaches one if $n \to \infty$.

[13]

For all chains which have an α-angle of a simple lock, at least so many cuts are required that each resulting chain has fewer edges than are needed for a simple lock to ensure that all resulting chain parts are not locked. All chain parts are examined in separated rooms, which does not take interlocks into account.

Combining the previous conjectures results in Conjecture 12 and Open Problem 11.

Conjecture 12:

For any α-angle $m_{lock}(\alpha, \frac{r}{l})$ exists that all models with $m(\alpha, \frac{r}{l}) \geq m_{lock}(\alpha, \frac{r}{l})$ are locked.

Open Problem 11:

Prove or disprove Conjecture 12. Try to prove this for chains and then extend to models. If Conjecture 12 is true, define the function $m_{lock}(\alpha, \frac{r}{l})$.

7. Conclusion

The main result is a reasoned but mainly unproved framework for locked chains. It is expected that, especially for closed chains, almost all chains are locked and if not locked these are rigid. Therefore, on the one hand, local configuration subspaces are smaller, on the other hand, it is not directly clear in almost every case whether two configurations are in the same configuration subspace. However, the given framework offers possibilities to determine this efficiently. Thus, essential configuration subspaces are identified. For open chains, it was initially assumed that these are never locked. It is shown that simple locks exist with the highest probability. It is also shown that the local configuration subspace of these simple locks has probably a convex hull. Therefore, an efficient reduction of the search space for protein backbones should be possible if these contain simple locks. Furthermore, it is assumed that all open chains with $n \to \infty$ are locked, but if this affects protein backbones or polymers is not clear. For further details have again a look at the summaries of open and closed chains.

8. Bibliography

[1] E. Demaine, M. Demaine, Y. Diomidov and K. Mundilova, "6.849: Geometric Folding Algorithms: Linkages, Origami, Polyhedra (Fall 2020)", MIT, 15 November 2010. [Online]. Available: http://courses.csail.mit.edu/6.849/fall20/lectures/L19.html. [Accessed 5. November 2021].

[2] E. Demaine and O. Joseph, "Open Problem 9.3: Locked Fixed-Angle Chains", in *Geometric Folding Algorithms: Linkages, Origami, Polyhedra*, Cambridge University Press, 2007, p. 154.

[3] K. C. Millett, "Physical knot theory: an introduction to the study of the influence of knotting on the spatial characteristics of polymers", 2011, doi:10.1142/9789814313001_0013.

[4] H. Rischer, G. Szilvay and K.-M. Oksman-Caldentey, "Cellular agriculture - industrial biotechnology for food and materials", *Current Opinion in Biotechnology,* vol. 61, pp. 128-134, 2020, doi:10.1016/j.copbio.2019.12.003.

[5] E. Demaine and J. O´Rourke, "Poducible Polygonal Protein Chains", in *Geometric Folding Algorithms: Linkages, Origami, Polyhedra*, Cambridge University Press, 2007, p. 148.

[6] N. Benbernou, "Fixed-Angle Polygonal Chains: Locked Chains and the Maximum Span", Smith College, Northampton, Massachusetts, 2006.

[7] B. Newland, Y. Zheng, Y. Jin, M. Abu-Rub, H. Cao, W. Wang and A. Pandit, "Single Cyclized Molecule Versus Single Branched Molecule: A Simple and Efficient 3D 'Knot' Polymer Structure for Nonviral Gene Delivery", *Journal of the American Chemical Society,* vol. 134, no. 10, p. 4782–4789, 2012.

[8] A. Aied, Y. Zheng, B. Newland and W. Wang, "Beyond Branching: Multiknot Structured Polymer for Gene Delivery", *Biomacromolecules,* vol. 15, no. 12, p. 4520–4527, 2014, doi:10.1021/bm5013162.

[9] B. Newland, A. Aied, A. V. Pinocely, Y. Zheng, T. Zhao, H. Zhang, R. Niemeier, E. Dowd, A. Pandit and W. Wang, "Untying a nanoscale knotted polymer structure to linear chains for efficient gene delivery in vitro and to the brain", *Nanoscale,* vol. 6, p. 7526–7533, 2014, doi:10.1039/c3nr06737h.

[10] "Oxford Reference: transfection", Oxford University Press, [Online]. Available: https://www.oxfordreference.com/view/10.1093/oi/authority.20110803105325967. [Accessed 10. January 2022].

[11] "Oxford Reference: in vitro", Oxford University Press, [Online]. Available: https://www.oxfordreference.com/view/10.1093/oi/authority.20110803100009938. [Accessed 10. January 2022].

[12] "Oxford Reference: In vivo", Oxford University Press, [Online]. Available: https://www.oxfordreference.com/view/10.1093/oi/authority.20110803100009956. [Accessed 10. January 2022].

[13] E. Demaine, S. Langerman and J. O´Rourke, "Geometric Restrictions on Producible Polygonal Protein Chains", *Algorithmica,* vol. 44, pp. 167-181, 2006, doi:10.1007/s00453-005-1205-7.

[14] E. Demaine and O. Joseph, "Open Problem 7.1: Unlocking Chains by Cutting", in *Geometric Folding Algorithms: Linkages, Origami, Polyhedra*, Cambridge University Press, 2007, p. 123.

[15] E. Demaine and J. O'Rourke, Geometric Folding Algorithms: Linkages, Origami, Polyhedra, Cambridge University Press, 2007.

[16] E. Demaine and S. Eisenstat, "Flattening Fixed-Angle Chains Is Strongly NP-Hard", in *Algorithms and Data Structures*, Berlin, Springer-Verlag, 2011, doi: 10.1007/978-3-642-22300-6_27, pp. 314-325.

[17] E. Demaine and J. O'Rourke, "Joint-Constraint Motion", in *Geometric Folding Algorithms: Linkages, Origami, Polyhedra*, Cambridge University Press, 2007, pp. 131,132.

[18] G. Aloupis, E. D. Demaine, V. Dujmović, J. Erickson, S. Langerman, H. Meijer, J. O'Rourke, M. Overmars, M. Soss, I. Streinu and G. Toussaint , "Flat-State Connectivity of Linkages under Dihedral Motions*", in *in Proceedings of the 13th Annual International Symposium on Algorithms and Computation (ISAAC 2002), Lecture Notes in Computer Science, volume 2518*, Vancouver, British Columbia, Canada, 2002.

[19] N. Benbernou and J. O'Rourke, "On the Maximum Span of Fixed-Angle Chains", in *Proceedings of the 18th Annual Canadian Conference on Computational Geometry*, Ontario, Canada, 2006.

[20] E. D. Demaine and J. O'Rourke, "Applications", in *Geometric Folding Algorithms: Linkages, Origami, Polyhedra*, cambridge university press, 2007, p. 11.

[21] L. H. Kauffman, "Knot Diagrammatics", 2008, doi:10.1016/B978-044451452-3/50007-1.

[22] R. Goldstein-Rose, "Introduction to Knots and Braids using Seifert Circles", 2007.

[23] A. Bosman, "Knot Theory 1: Coloring", Andrews University, 9 1 2019. [Online]. Available: https://www.youtube.com/watch?v=fwcvmo0y_SI. [Accessed 17. December 2021].

[24] K. Reidemeister, Knotentheorie, Berlin, Heidelberg: Springer, 1932.

[25] H. Schubert, Die eindeutige Zerlegbarkeit eines Knotens in Primknoten, Heidelberg, Deutschland: Springer, Berlin, Heidelberg, 1949, doi:10.1007/978-3-642-45813-2.

[26] W. W. Eric, "Reidemeister Moves", Wolfram Research, [Online]. Available: https://mathworld.wolfram.com/ReidemeisterMoves.html. [Accessed 17. December 2021].

[27] J. Przytycki, "3-coloring and other elementary invariants of knots", *Banach Center Publications,* vol. 42, pp. 275-295, 1998.

[28] C. Lescop, "Knot Invariants and Configuration Space Integrals", 2005, doi:10.1007/11374060_1.

[29] V. F. R. Jones, "Knot Theory and Statistical Mechanics", *Scientific American ,,* vol. 263, no. 5, pp. 98-105, 1990.

[30] K. C. Millett, "Knotting of Regular Polygons in 3-Space", *Journal of Knot Theory and Its Ramifications,* vol. 3, no. 3, pp. 263-278, 1994.

[31] T. Biedl, E. Demaine, M. Demaine, S. Lazard, A. Lubiw, J. O'Rourke, M. Overmars, S. Robbins, I. Streinu, G. Toussaint and S. Whitesides, "Locked and Unlocked Polygonal Chains in 3D", *Discrete & Computational Geometry,* vol. 26, pp. 269-281, 2001, doi:10.1007/s00454-001-0038-7.

[32] R. Connelly, E. D. Demaine and G. Rote, "Infinitesimally locked self-touching linkages with applications to locked trees", in *Physical Knots: Knotting, Linking, and Folding of Geometric Objects in 3-space*, American Mathematical Society, 2002, pp. 287-311.

[33] H. Alt, C. Knauer, G. Rote and S. Whitesides, "On the Complexity of the Linkage Reconfiguration Problem", 2003, doi:10.1090/conm/342/06126.

[34] G. Aloupis, E. Demaine, H. Meijer, J. O´Rourke, S. Ileana and G. Toussaint, "On Flat-State Connectivity of Chains with Fixed Acute Angles", in *Proceedings of the 14th Canadian Conference on Computational Geometry (CCCG 2002)*, Lethbridge, 2002.

[35] M. Soss and G. Toussaint, "Geometric and computational aspects of polymer reconfiguration," *Journal of Mathematical Chemistry,* vol. 27, p. 303–318, 2000, doi:10.1023/A:1018823806289.

[36] E. D. Demaine and J. O'Rourke, "Classification", in *Geometric Folding: Linkages, Origami, Polyhedra*, Cambridge University Press, 2007, pp. 10-11.

[37] E. D. Demaine and J. O'Rourke, "General Algorithms and Upper Bounds, Probabilistic Roadmaps", in *Geometric Folding Algorithms: Linkages, Origami, Polyhedra*, Cambridge University Press, 2007, pp. 17-19, 154-156.

[38] L. E. Kavraki and S. M. LaValle, "Motion Planning", 2007.

[39] L. E. Kavraki and S. M. LaValle, "Motion Planning", 2008.

[40] J. Lummerzheim, "Inversion, ring formation and randomness testing of different extensions of the Langton's ant algorithm on multidimensional tori for the creation of Langton's ant-based cryptosystems", 2021, urn:nbn:de:hbz:832-epub4-16978.

[41] D. A. Joseph and W. H. Plantinga, "On the complexity of reachability and motion planning questions", in *Proceedings of the first annual symposium on computational geometry*, Baltimore, Maryland, USA, 1985, doi:10.1145/323233.323242.

[42] J. Canny, "A new algebraic method for robot motion planning and real geometry", in *28th Annual Symposium on Foundations of Computer Science (sfcs 1987)*, Los Angeles, CA, USA, 1987, doi:10.1109/SFCS.1987.1.

[43] G. E. Collins, "Quantifier elimination for real closed fields by cylindrical algebraic decompostion", *Preliminary Report, Proceedings of EUROSAM 74, STGSAM Bulletin,* vol. 8, no. 3, p. 80–90, 1974.

[44] E. Demaine and J. O´Rourke, "History", in *Geometric Folding Algorithms: Linkages, Origami, Polyhedra*, Cambridge University Press, 2007, p. 88.