universität
wien

MASTERARBEIT / MASTER'S THESIS

Titel der Masterarbeit / Title of the Master's Thesis

Angle rigidity of $\mathbb{Z}^2$ configurations in $\mathbb{R}^3$

verfasst von / submitted by

Andrii Dubkov, BSc

in partial fulfilment of the requirements for the degree of

Master of Science (MSc)

Wien, 2020 / Vienna, 2020

Studienkennzahl lt. Studienblatt / degree programme code as it appears on the student record sheet:	UA 066 821
Studienrichtung lt. Studienblatt / degree programme as it appears on the student record sheet:	Masterstudium Mathematik UG2002
Betreut von / Supervisor:	Univ.-Prof. Ulisse Stefanelli, PhD

Abstract

The goal of this thesis is to investigate subconfigurations of the 2–dimensional square lattice $\mathbb{Z}^2$ and to investigate how they may be deformed in the 3–dimensional space $\mathbb{R}^3$. We especially address some of those configurations, which can be deformed in $\mathbb{R}^3$ without changing their bond lengths and bond angles. Necessary and sufficient conditions related to such a property are discussed. We also conduct a computational experiment and discuss the results.

Zusammenfassung

Das Ziel dieser Arbeit ist es, Unterkonfigurationen des zweidimensionalen quadratischen Gitters $\mathbb{Z}^2$ zu untersuchen und auch zu untersuchen, wie sie im dreidimensionalen Raum $\mathbb{R}^3$ deformiert werden können. Wir befassen uns insbesondere mit einigen dieser Konfigurationen, die im $\mathbb{R}^3$ deformiert werden können, ohne ihre Bindungslängen und Bindungswinkel zu ändern. Notwendige und ausreichende Bedingungen im Zusammenhang mit einer solchen Eigenschaft werden diskutiert. Wir führen auch ein Computerexperiment durch und diskutieren die Ergebnisse.

Contents

1 Motivation and overview

The understanding of the properties of 2–dimensional point configurations in 3–dimensional space is very important in both applied and theoretical situations. In this thesis we present some subconfigurations of $\mathbb{Z}^2$ and investigate, how they deform in $\mathbb{R}^3$.

The thesis is structured in the following way:

At the beginning we make a quick recap of some important graph-theoretical definitions, which will be useful later (Section 2.1). As a sidenote, we also show a couple of important theorems, which (among other possibilities) can be solved with the help of graph theory — this confirms, that graph theory finds its application in many different areas of mathematics. Then, we address the notion of rigidity (Section 2.2) and go deeper into the history of it by looking at Laman graphs (Section 2.2.1), which are central to the study of rigid graphs. We continue our preparation for the main part by defining the energy of point configurations (Section 2.2.2).

In the main part (Section 3), we firstly define angle-rigid configurations (Section 3.1), where bond lengths and bond angles change under any nontrivial deformations (see Figure 6). Next we show, that angle-rigid configurations are strict local energy minimizers in Proposition 3.4.

In this thesis we are interested in *not angle-rigid* configurations. As we have mentioned before, we are going to look only at some specific subconfigurations of $\mathbb{Z}^2$. Therefore, we need to specify, which configurations are we interested in. We call these configurations *special not angle-rigid configurations* (Section 3.2). We show, that these special not angle-rigid configurations can be further subcategorized into so-called *"book-flip"* and *"torque"* types. We also give an example of a not angle-rigid configuration, which cannot be classified as a *special* one (Figure 12).

As the main result of the thesis (Section 3.3) we state and prove two theorems, that characterize not angle-rigid configurations.

Finally, in Section 4 we take the computer's aid to investigate, how common the special not angle-rigid configurations are. We give the explanation of chosen parameters (Section 4.1), then we describe the used algorithm in words and by using a pseudocode (Section 4.2). The computational results are in Section 4.3, where we also present some conclusions and conjectures. The code may be found in the Appendix (Section A).

2 Prerequisites

2.1 Basic definitions and properties

In this section we would like to recap some basic graph-theoretical definitions.

A *graph* is a pair $G = (V, E)$ of two finite sets V and E: The elements of V are called *vertices*, and the elements of E are called *edges*. There are some specific edges we would like to mention: a *loop* is an edge that joins a single endpoint to itself, and a *multi-edge* is a collection of two or more edges having identical endpoints.

In Figure 1 we can see an example of graphs with a loop and a multi-edge:

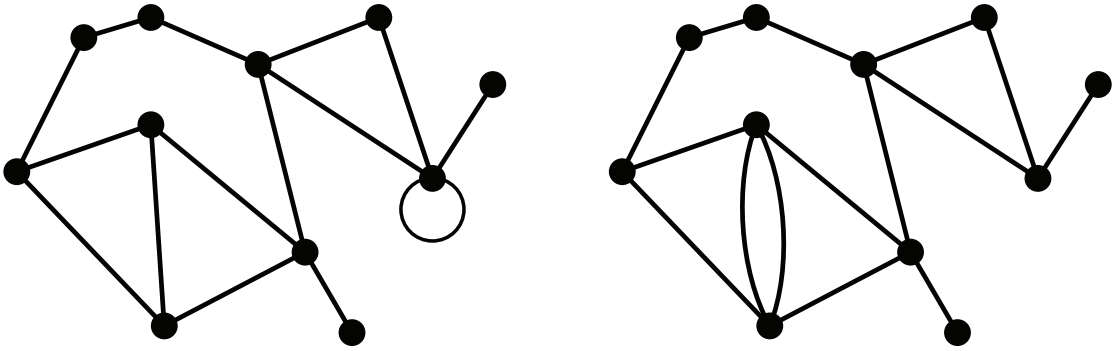

Figure 1: An example of graphs with a loop and a multi-edge.

A *simple graph* is a graph that has neither loops nor multi-edges.

A *planar graph* is a graph that can be embedded in the plane, i.e. it can be drawn on the plane in such a way that its edges intersect only at their endpoints. In other words, it can be drawn in such a way that no edges cross each other. In Figure 2 we can see an example of a planar graph and a non-planar one:

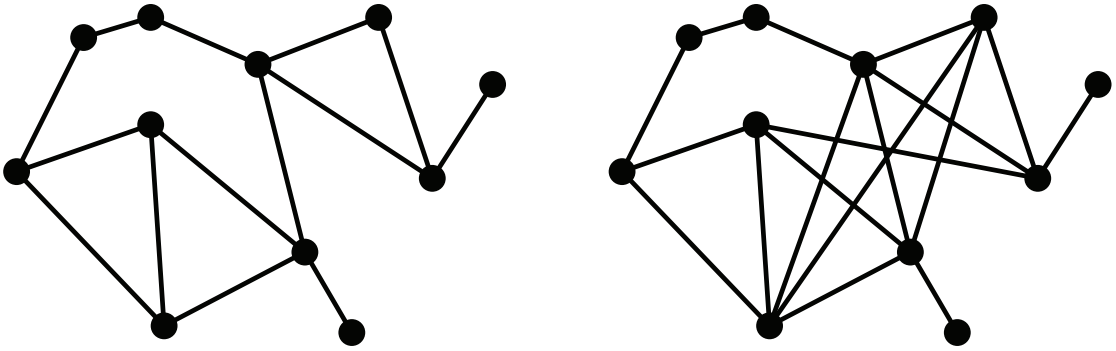

Figure 2: An example of planar and non-planar graphs.

In a graph G, a *walk* from vertex v_0 to vertex v_n is an alternating sequence

$$v_0, e_1, v_1, e_2, \ldots, v_{n-1}, e_n, v_n$$

of vertices v_i and edges e_i, such that endpoints of e_i are v_{i-1} and v_i, for $i = 1, \ldots, n$. The vertex v is *reachable* from vertex u if there is a walk from u to v. A graph is *connected* if for every pair of vertices u and v there is a walk from u to v.

A *bipartite* graph G is a graph whose vertices can be divided into two disjoint and independent sets A and B such that every edge connects a vertex in A to one in B. We denote such graphs by $G(A, B; E)$.

A vertex is *incident* to an edge if the vertex is one of the two vertices the edge connects. A *degree of a vertex* of a graph is the number of edges that are incident to the vertex; loops are counted twice. A graph is called *sparse* if it has a very low number of edges.

Now we state a couple of problems, which can be solved using graph theory.

Theorem 2.1 (Fermat's Little Theorem)**.** *Let $a \in \mathbb{N}$ and p be a prime number, such that p does not divide a. Then $a^p - a$ is divisible by p.*

Proof. [9] Consider the graph $G = (V, E)$, where the vertex set V is the set of all sequences $(a_1, a_2, \ldots, a_p)$ of natural numbers between 1 and a (included), with $a_i \neq a_j$ for some $i \neq j$. Clearly, V has $a^p - a$ elements. If

$$u = (u_1, u_2, \ldots, u_p),\ v = (u_p, u1, \ldots, u_{p-1}) \in V,$$

we say $\{u, v\} \in E$. With this assumption, each vertex of G is of degree 2. So, each component of G is a cycle of length p. Therefore, the number of components is $\frac{a^p - a}{p}$. That is, p divides $(a^p - a)$. □

Theorem 2.2 (Cantor-Schröder-Bernstein)**.** *Let A and B be two sets. If there is an injective mapping $f : A \to B$ and an injective mapping $g : B \to A$, then there is a bijection between A and B.*

Proof. [12] Without loss of generality, we may assume that $A \cap B = \varnothing$. Define a bipartite graph $G = (A, B; E)$, where $x, y \in E$ if and only if either $f(x) = y$ or $g(y) = x$, for $x \in A$ and $y \in B$. By our hypothesis, $1 \leqslant deg(v) \leqslant 2$ for each vertex v of G. Therefore, each component of G is either a one-way infinite path (i.e. a path of the form $x_0, x_1, \ldots, x_n$), or a cycle of even length with more than two vertices, or an edge. Note that a finite path of length at least 2 cannot be a component of G. Hence, in each component there is a set of edges such that each vertex in the component is incident with precisely one of these edges. Hence, in each component, the subset of vertices from A is of the same cardinality as the subset of vertices from B. □

2.2 Rigidity

Laman graphs are graphs belonging to the family of sparse graphs, which help to describe minimally rigid systems of bars and joints on a plane. These graphs are named after professor of the University of Amsterdam Gerard Laman, who used these graphs in 1970 to describe planar rigid structures. Laman graphs appear in the rigidity theory and are minimal rigid graphs.

Before Laman wrote about these graphs, Lebrecht Henneberg described them in a different way, and showed 2 operations, which can be used to construct such graphs: see Figures 4 and 5.

2.2.1 Laman Graphs

In this section we assume the graphs to be finite (i.e $|V| < \infty$), undirected, connected and simple.

A *labeling* of a graph $G = (V, E)$ is a function $L : E \to \mathbb{R}$. The pair (G, L) is called a *labeled graph*. A *realization* of $G = (V, E)$ is a function $r : V \to \mathbb{R}^2$. A realization r is said to be *compatible* with labeling L, if $\forall e \in E\ L(e) = ||r(a) - r(b)||_2$, where $a, b \in V$ are endpoints of e. A labeled graph (G, L) is *realizable* if and only if there exists a realization r, which is compatible with L.

A *direct Euclidean isometry* is an affine-linear map $\mathbb{R}^2 \to \mathbb{R}^2$ that preserves the distance and orientation in $\mathbb{R}^2$. We say that two realizations r_1 and r_2 of a graph G are *equivalent* if and only if there exists a direct Euclidean isometry σ of $\mathbb{R}^2$ such that $r_1 = \sigma \circ r_2$.

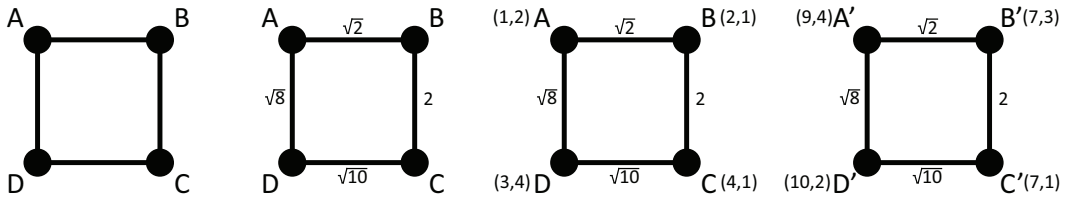

Figure 3: A graph G; a graph G with a labeling; a graph G with a realization, compatible with a given labeling; a graph G with a realization, equivalent to the previous realization.

A labeled graph (G, L) is called *rigid*, if it satisfies:

- (G, L) is realizable,
- there are only finitely many realizations compatible with L, up to equivalence.

Notation 2.3. For a graph $G = (V, E)$ the set of all possible labelings $L : E \to \mathbb{C}$ forms a vector space, denoted $\mathbb{C}^E$. Since $\mathbb{C}^E$ is a vector space, it makes sense to discuss the properties of a general labeling: a property P holds for a general labeling if $\{L \in \mathbb{C}^E : P(L) \text{ does not hold}\}$ is contained in an algebraic proper subset of $\mathbb{C}^E$.

Definition 2.4. A graph G is called *generically realizable* if for a general labeling L the labeled graph (G, L) is realizable.

A graph G is called *generically rigid* if for a general labeling L the labeled graph (G, L) is rigid.

Remark 2.5. If a graph G is generically realizable, so is every subgraph of G. In fact, every general labeling L' for $G' \subset G$ can be extended to a general labeling L for G, and since by the latter has a compatible realization, the former admits one as well.

Definition 2.6. A Laman subgraph of a graph $G = (V, E)$ is a graph $G' = (V', E')$ such that $V' \subset V, |V'| \geqslant 2$ and $E' = E|_{V'} := \{e_{ij} \in E : v_i, v_j \in V'\}$.

A Laman graph is a graph $G = (V, E)$ with $|E| = 2|V| - 3$ and $|E'| \leqslant 2|V'| - 3$ $\forall G' \subset G$ with $G' = (V', E')$.

Definition 2.7. For algebraic sets X, Y, a dominant map is a map $f : X \to Y$, where $Y \backslash f(X)$ is contained in an algebraic proper subset of Y.

For any graph $G = (V, E)$, there is a natural map r_G from the set $\mathbb{C}^{2|V|}$ of its realizations to the set $\mathbb{C}^E$ of its labelings:

$$r_G : \mathbb{C}^{2|V|} \to \mathbb{C}^E,$$

$$(x_v, y_v)_{v \in V} \mapsto ((x_u - x_v)^2 + (y_u - y_v)^2)_{u,v \in E}.$$

Each fiber of r_G, i.e. a preimage $r_G^{-1}(p)$ of a single point $p \in \mathbb{C}^E$, is invariant under the group of direct complex isometries. We define a subspace $\mathbb{C}^{2|V|-3} \subseteq \mathbb{C}^{2|V|}$ as follows: choose two distinguished vertices $\overline{u}$ and $\overline{v}$ with $\overline{u}, \overline{v} \in E$, and consider the linear subspace defined by the equations $x_{\overline{u}} = y_{\overline{u}} = 0$ and $x_{\overline{v}} = 0$. In this way the subspace $\mathbb{C}^{2|V|-3}$ intersects every orbit of the action of isometries on a fiber of r_G in exactly two points: in fact, the equations do not allow any further translation or rotation; however, for any labeling $\lambda : E \to \mathbb{C}$ and for every realization in $\mathbb{C}^{2|V|-3}$ compatible with λ there exists another realization, obtained by multiplying the first one by -1, which is equivalent, but gives a different point in $\mathbb{C}^{2|V|-3}$. The restriction of r_G to $\mathbb{C}^{2|V|-3}$ gives the map $h_G : \mathbb{C}^{2|V|-3} \to \mathbb{C}^E$.

The following statement follows from the construction of h_G; notice that the choice of $\overline{u}$ and $\overline{v}$ has no influence on the result.[1]

Proposition 2.8. *[1] A graph G is generically rigid if and only if h_G is dominant and a general fiber of h_G is finite. This is equivalent to saying that h_G is dominant and $2|V| = |E| + 3$.*

Proof. [1] It is enough to notice that if h_G is dominant, then the dimension of the general fiber is $2|V| - 3 - |E|$. □

Theorem 2.9. *[1] Let $G = (V, E)$ be a graph. Then the following three conditions are equivalent:*

1. *G is a Laman graph;*
2. *G is generically rigid;*
3. *G can be constructed by iterating the two Henneberg rules (see Figures 4 and 5), starting from the graph that consists of two vertices connected by an edge.*

Figure 4: The first Henneberg rule: given any two vertices a and b (which may be connected by an edge or not), we add a vertex c and the two edges $\{a, c\}$ and $\{b, c\}$.

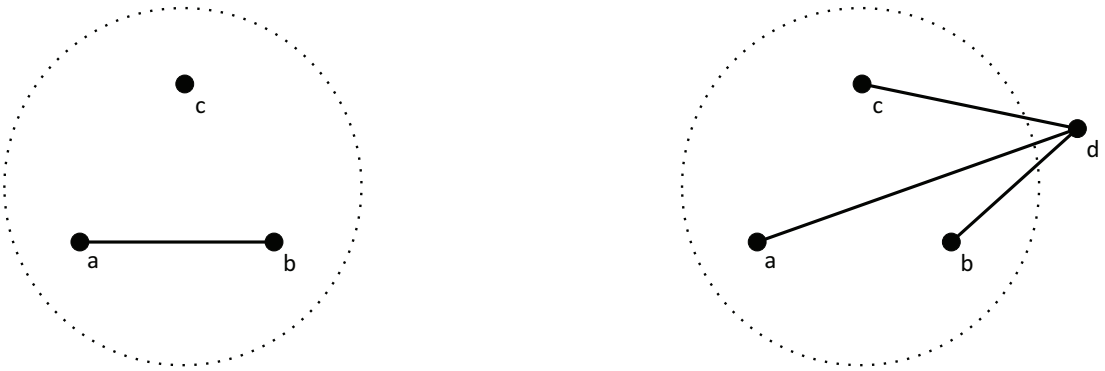

Figure 5: The second Henneberg rule: given any three vertices a, b and c such that a and b are connected by an edge, we remove the edge $\{a, b\}$, we add a vertex d and the three edges $\{a, d\}$, $\{b, d\}$ and $\{c, d\}$.

Proof. 2 $\implies$ 1: Assume that G is generically rigid. Then every subgraph $G' = (V', E')$ is generically realizable (see Remark 2.5), and so the map $h_{G'}$ is dominant.

Therefore, the dimension of the codomain is bounded by the dimension of the domain, which says $2|V'| - 3 \geqslant |E'|$. The equality in the previous formula for the whole graph G follows from Proposition 2.8.

1 $\implies$ 3: We prove the statement by induction on the number of vertices. The induction base with two vertices is clear. Assume that G is a Laman graph with at least 3 vertices. By [6], Proposition 6.1, the graph G has a vertex of degree 2 or 3. If G has a vertex of degree 2, then the subgraph G' obtained by removing this vertex and its two adjacent edges is a Laman graph by [6], Theorem 6.3. By induction hypothesis, G' can be constructed by Henneberg rules, and then G can be constructed from G' by the first Henneberg rule. Assume now that G has a vertex v of degree 3. By [6], Theorem 6.4, there are two vertices u and w connected with v such that the graph G' obtained

by removing v and its three adjacent edges and then adding the edge $\{u, w\}$ is Laman. By induction hypothesis, G' can be constructed by Henneberg rules, and then G can be constructed from G' by the second Henneberg rule.

3 $\implies$ 2: We prove the statement by induction on the number of Henneberg rules. The induction base is the case of the one-edge graph, which is generically rigid. By induction hypothesis we assume that $G = (V, E)$ is generically rigid.

Perform a Henneberg rule on G and let G' be the result. We intend to show that G' is generically rigid, too.

As far as the first Henneberg rule is concerned, we observe that for any realization of G, compatible with a general labeling L, and for any labeling L' extending L we can always construct exactly two realizations of G' that are compatible with L'.

Let us now assume that G' is constructed with the second Henneberg rule. Call t the new vertex of G', and denote the three vertices to which it is connected by u, v and w. Let G'' be the graph obtained by removing from G the same edge e that is removed in G'. Without loss of generality we assume $e = \{u, v\}$. We first show that G' is generically realizable. Let $N : \mathbb{C}^2 \to \mathbb{C}$ be the quadratic form corresponding to the bilinear form $\langle \cdot, \cdot \rangle$.

Fix a general labeling for G''. We define the algebraic set $C \subseteq \mathbb{C}^3$ as the set of all points (a, b, c) such that there is a compatible realization r of G'' satisfying

$$N(r(u) - r(v)) = a, N(r(u) - r(w)) = c, N(r(v) - r(w)) = b.$$

For a general $a_0 \in \mathbb{C}$ there exist finitely many (up to equivalence) points $(a_0, b, c) \in C$, namely the "lengths" of the triangle (u, v, w) that come from the finitely many realizations of G. It follows that $dim(C) \geqslant 1$.

A complex version of a classical result in distance geometry (see [3], Theorem 2.4) states that four points $p_0, p_1, p_2, p_3 \in \mathbb{C}^2$ fulfill

$$N(p_0 - p_1) = x, N(p_0 - p_2) = y, N(p_0 - p_3) = z,$$

$$N(p_1 - p_2) = a, N(p_2 - p_3) = b, N(p_1 - p_3) = c,$$

if and only if the following Cayley-Menger determinant

$$F(a, b, c, x, y, z) := det \begin{bmatrix} 0 & a & c & x & 1 \\ a & 0 & b & y & 1 \\ c & b & 0 & z & 1 \\ x & y & z & 0 & 1 \\ 1 & 1 & 1 & 1 & 0 \end{bmatrix}$$

vanishes. We define

$$U := \bigcup_{p=(a,b,c) \in C} S_p,$$

where

$$S_p := \{(x, y, z) \in \mathbb{C}^3 : F(x, y, z, a, b, c) = 0\}$$

and

$$e_p := (abc : a(a-b-c) : b(b-a-c) : c(c-a-b) : a-b-c : b-a-c : c-a-b : a : b : c) \in \mathbb{P}^9_{\mathbb{C}}.$$

The point e_p in $\mathbb{P}^9_{\mathbb{C}}$ with $p = (a, b, c)$ has coordinates given by the coefficients of $F(a, b, c, x, y, z)$, considered as a polynomial in x, y and z. Because of this, the point e_p determines S_p uniquely as a surface. The function $\mathbb{C}^3\backslash\{0\} \to \mathbb{P}^9_{\mathbb{C}}$ sending $p \mapsto e_p$ is injective, and hence the family $(S_p)_{p\in C}$ of surfaces is not constant. It follows that the algebraic set U has dimension 3, and thus a general point $(x, y, z) \in \mathbb{C}^3$ lies in U.

If we extend the general labeling of G'' by assigning a general triple $(x, y, z) \in U$ as labels to the three new edges, then we get at least one realization of G'. It follows that G' is generically realizable.

Since $|V'| = |V| + 1$ and $|E'| = |E| + 2$, it follows that $2|V| = |E'| + 3$. Since, as we have just shown, the map $h_{G'}$ is dominant, the graph G' is generically rigid by Proposition 2.8. □

2.2.2 Energy

Definition 2.10. A configuration C is a subset of $\mathbb{Z}^2$.

In this thesis, we will look at the behavior of some subconfigurations of the square lattice in the 3–dimensional space $\mathbb{R}^3$.

Let $\varepsilon \in (0, \frac{\sqrt{2}-1}{4})$ be fixed. To each collection $X = \{x_k\}_{k=1}^N$ of N points in $\mathbb{R}^3$ we associate the set of *neighbors* by the defining the pairs

$$N(X) = \{(x_k, x_{k'}) \in X^2 \ : \ 1 - \varepsilon < |x_k - x_{k'}| < 1 + \varepsilon\}.$$

Note that each pair of neighbors is counted twice in $N(X)$.

Bonds are identified with pairs $(x_k, x_{k'}) \in N(X)$, up to permutations. We associate to X the respective *bond graph* resulting from taking points as vertices and bonds as edges.

We shall mainly use *point* and *bond* for *vertex* and *edge* in the following, still resorting to the graph-theory terminology in specific places. In particular, we say that a configuration is *connected* if the corresponding bond graph is connected.

We also define a set of *triplets* of neighbors in X by letting

$$T(X) = \{(x_k, x_{k'}, x_{k''}) \in X^3 \ : \ (x_k, x_{k'}),\ (x_{k'}, x_{k''}) \in N(X),\ x_k \neq x_{k''}\}.$$

To each triplet $(x_k, x_{k'}, x_{k''}) \in T(X)$ we uniquely associate the angle $\theta(x_k, x_{k'}, x_{k''})$ formed by the segments $x_k - x_{k'}$ and $x_{k''} - x_{k'}$ and oriented clockwise.

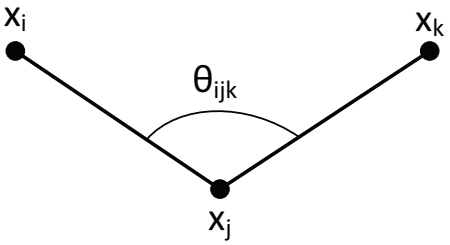

Figure 6: Notation for angles and bonds.

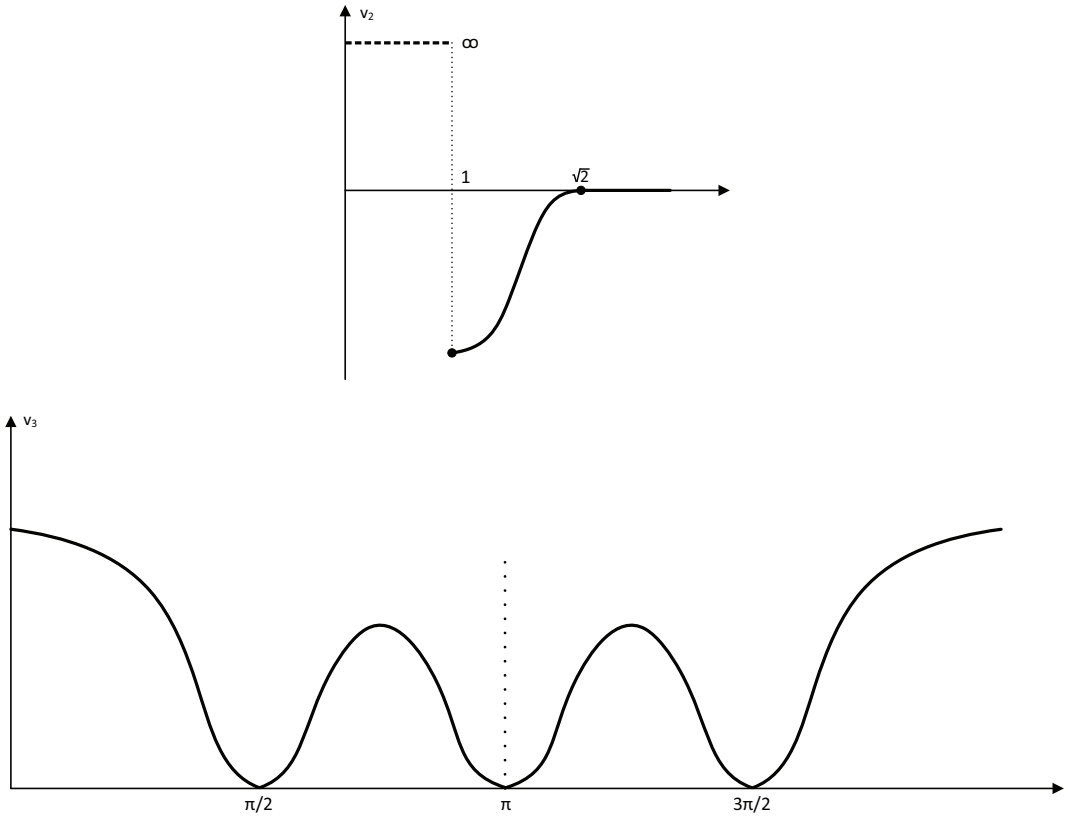

Figure 7: Examples of energy functions v_2 and v_3.

All triplets are, again, counted twice in $T(X)$.

To each $X = \{x_k\}_{k=1}^N \in \mathbb{R}^{3N}$ we associate the *configurational energy*

$$E : \mathbb{R}^{3N} \to \mathbb{R} \cup \{\infty\}$$

given by

$$E(X) = E_2(X) + E_3(X) := \frac{1}{2} \sum_{N(X)} v_2(|x_k - x_{k'}|) + \frac{1}{2} \sum_{T(X)} v_3(\theta(x_k, x_{k'}, x_{k''})).$$

The configurational energy is the sum of a *two-body* term E_2, depending solely on the mutual distance of the points, and a *three-body* contribution E_3 depending on angles instead. The two-body *interaction density* $v_2 : [0, \infty) \to [0, \infty]$ with $v_2(l) = 0 \Leftrightarrow l = 1$ is assumed to be of attractive-repulsive type, short-ranged and strictly minimized at the reference distance 1. The three-body interaction energy E_3 is modulated via the *three-body* interaction density $v_3 : [0, 2\pi] \to [0, \infty)$ which we assume to be symmetric around π and attain its minimal value 0 just at $\pi/2$, π, and $3\pi/2$.

We also assume in the following, that $v_3(\theta) = v_3(2\pi - \theta)\ \forall \theta \in [0, 2\pi)$.

The energy E being invariant by translation and rotation, we shall tacitly assume, that all statements in the following are to be considered up to isometries. Note incidentally that E is coercive. Under lower semincontinuity assumptions on the interaction densities the existence of global minimizers immediately follows. Still, the exact geometry of global minimizers is mostly not known.

Square configurations (C_n, see [7]) are global minimizers (ground states) in a plane , i.e. they have the lowest energy among other configurations with similar topological characteristics. These configurations can be looked at as molecules or crystals. Some properties of the molecules on the hexagonal lattice or crystals on the square lattice were investigated in [7, 11]. Also in [2] the authors investigate some properties of carbon nanotubes, which depend on the radius of the nanotube and the way it is rolled up from a graphene sheet. This, as appears, introduces some changes to the bond angles and bond lengths, which is crucial for devising accurate mechanical models.

3 Main part

3.1 Angle rigidity

Notation 3.1. We interpret $\mathbb{Z}^2$ as a proper subset of $\mathbb{R}^3$, namely,

$$\mathbb{Z}^2 = \{(i, j, 0) \ : \ i, \, j \in \mathbb{Z}\}.$$

In order to shorten notation, whenever possible we refer to points in $\mathbb{Z}^2$ by specifying their first two integer coordinates only. We call *axes* the subsets of $\mathbb{Z}^2$ the form $\{i\} \times \mathbb{Z}$ or $\mathbb{Z} \times \{j\}$, for $i, \, j \in \mathbb{Z}$.

Note that all configurations C of N point in $\mathbb{Z}^2$ (indicated as $C \subset \mathbb{Z}^{2N}$ in the following) are global minimizers of the energy E as $E(C) = 0$.

Definition 3.2. We call a continuous mapping $\varphi : [0, 1] \times C \to \mathbb{R}^{3N}$, $(t, x) \mapsto \varphi_t(x)$, a *small deformation* of C if $\varphi_0(C) = C$ and

$$\max_{t \in [0,1]} \sum_{k=1}^{N} |x_k - \varphi_t(x_k)| < \varepsilon.$$

In particular, if φ is a small deformation of C we have that $N(\varphi_t(C)) = N(C)$ and $T(\varphi_t(C)) = \varphi(C)$ for all $t \in [0, 1]$. Namely, the topology of the bond graph of C and $\varphi_t(C)$ is the same for all $t \in [0, 1]$.

Definition 3.3 (Angle-rigid configuration)**.** We say that a configuration $C \subset \mathbb{Z}^{2N}$ is *angle-rigid* if there exists no nontrivial small deformation of C leaving bond lengths and angles unchanged. Namely, if φ is a small deformation of C fulfilling for all $t \in [0,1]$

$$|\varphi_t(x_k) - \varphi_t(x_{k'})| = 1 \quad \forall (x_k, x_{k'}) \in N(C), \tag{1}$$

$$\theta(\varphi_t(x_k), \varphi_t(x_{k'}), \varphi_t(x_{k''})) = \theta(x_k, x_{k'}, x_{k''}) \quad \forall (x_k, x_{k'}, x_{k''}) \in T(C), \tag{2}$$

then φ is necessarily the identity, namely, $\varphi_t(x) = x$ for all $t \in [0,1]$.

The gist of the latter definition is the following observation.

Proposition 3.4 (Angle-rigidity = strict local minimality)**.** *A configuration is a strict local minimizer for the energy E if and only if it is angle-rigid.*

Proof. Let $C = \{x_k\}_{k=1}^N \in \mathbb{Z}^{2N}$ be angle-rigid and $\widetilde{C} = \varphi_1(C) = \{\widetilde{x}_k\}_{k=1}^N \in \mathbb{R}^{3N}$ where φ is a small deformation of C. Since C is angle-rigid, if $\widetilde{C} \neq C$ there exists $(x_k, x_{k'}) \in N(C)$ with

$$|\varphi_t(x_k) - \varphi_t(x_{k'})| \neq 1$$

for some $t \in (0,1]$ or there exists $(x_k, x_{k'}, x_{k''}) \in T(X)$ with

$$\theta(\varphi_{t'}(x_k), \varphi_{t'}(x_{k'}), \varphi_{t'}(x_{k''})) \notin \{\pi/2, \pi, 3\pi/2\}$$

for some $t' \in (0,1]$ (or both). In all cases $E(\widetilde{C}) > 0 = E(C)$ which proves that C is a strict local minimizer. Let now C be not angle-rigid. Then there exists a nontrivial small deformation φ of C such that (1)-(2) hold. By defining $\widetilde{C} = \varphi_t(C)$ for $t \in (0,1]$ small enough we have that $E(\widetilde{C}) = E(C)$, so that C is not a strict local minimizer. □

All disconnected configurations are obviously not angle-rigid. Among connected configurations there are angle-rigid and not angle-rigid configurations.

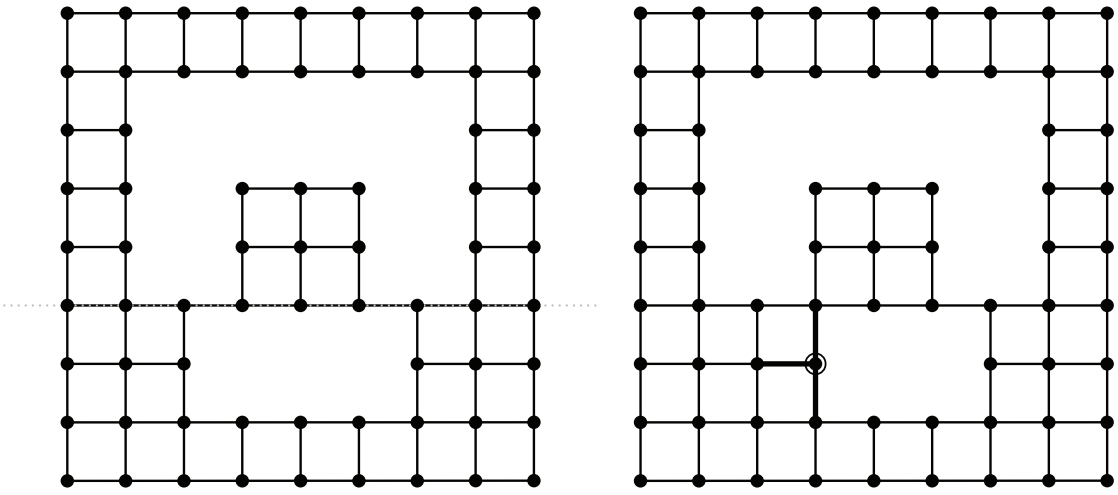

Figure 8: A connected (special – see the definition below 3.5) not angle-rigid configuration (left) and connected angle-rigid configuration (right).

In the example above (Figure 8) we took a special (Definition 3.5) not angle-rigid configuration by a book-flip and added one extra point (circled on the right configuration). This produces 3 extra bonds (drawn bold on the right configuration), which "lock" the configuration, disabling it from deforming not angle-rigidly.

3.2 Special non-angle-rigidity

Checking angle-rigidity for connected configurations from the definition may be very involved. We are hence interested in finding more tractable conditions implying angle-rigidity. In 3–dimensional space there are examples of configurations, which can be deformed in 3–dimensional space, while their bonds and angles remain unchanged. There are several distinct subcategories of these deformations, but here we will take a closer look at 2 main subcategories:

In the first case some connected subset of points can leave the original plane, by moving "upwards" (*wlog* we consider our original configuration in the x-y–plane, where the move "upwards" is considered to be in the positive direction of the z–axis.) — we call this type of deformation *book-flip*.

In the other case some connected subset of points can move "upwards" on one side and "downwards" on the other side - we call this type of deformation *torque*.

It seems rather obvious, that both these deformations can be thought of as some kind of rotations. Therefore it makes sense to define a rotational axis, which remains in the original plane, and the whole action happens around it. If we look a bit closer at our configuration, which appears to be not angle-rigid in $3-$dimensional space, we can see, that the part, which can be rotated about some axis, is not connected to the other points of the configuration on at least one side of the axis. Therefore it's a perfect time to state the following

Definition 3.5 (Special not angle-rigid configuration)**.** We say that a configuration $C \subset \mathbb{Z}^{2N}$ is *a special not angle-rigid configuration* if there exists an axis α and a sub-configuration $\widetilde{C} \subset C$ such that both $\widetilde{C} \backslash \alpha$ and $(C \backslash \widetilde{C}) \backslash \alpha$ are not empty and, letting $R(t)$ be a rotation about α of amplitude t/θ with $\theta \in (0, \pi/2)$, the mapping

$$\varphi_t(x) = \begin{cases} R(t)x & \text{if } x \in \widetilde{C} \\ x & \text{if } x \in C \backslash \widetilde{C} \end{cases}$$

leaves bonds unchanged, namely fulfills (1).

Remark 3.6. Note that, although not specifically mentioned in the definition above, a *special not angle-rigid configuration* is actually not angle-rigid, for the same deformation φ_t preserves angles as well, namely (2) holds.

In Figure 9 we can see some examples of configurations, that we call *special not angle-rigid configurations*. The rotational axis α is represented by a dotted gray line.

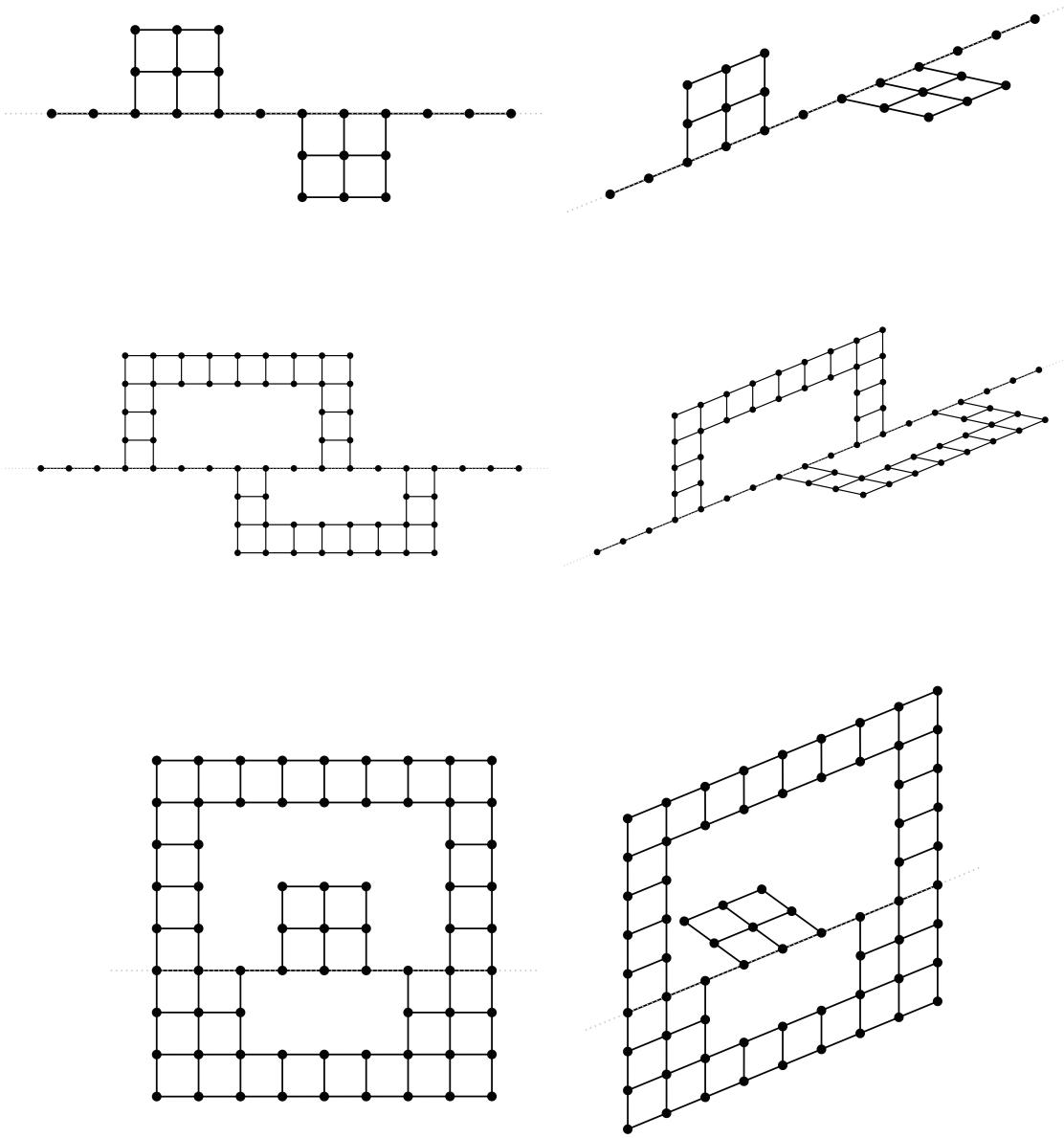

Figure 9: Several examples of not angle-rigid configurations by a book-flip.

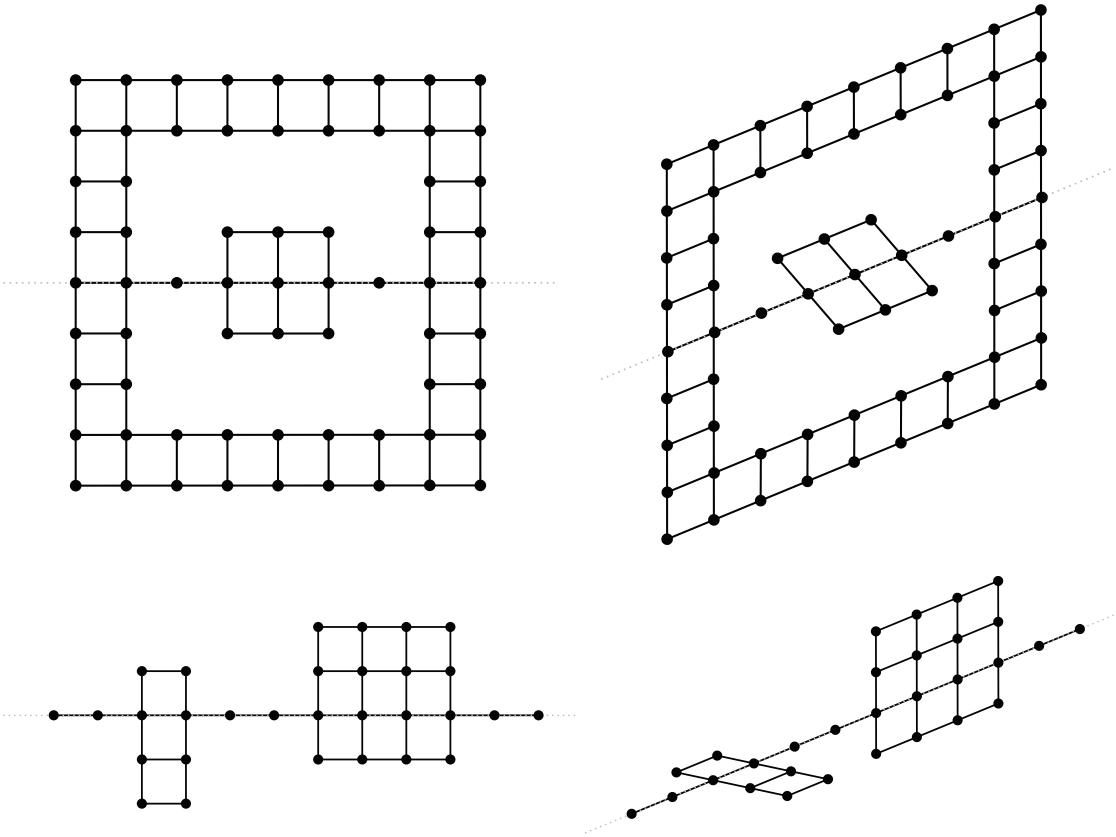

Figure 10: Several examples of not angle-rigid configurations by a torque.

Here we can see a configuration, that has both book-flip- and torque axes. The book-flip axis is a horizontal one, and the torque axis is a vertical one.

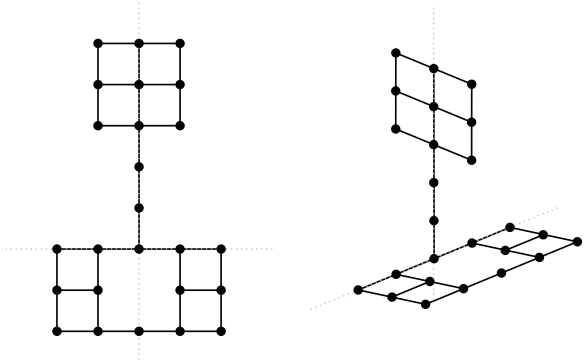

Figure 11: An example of a combination of both book-flip and torque in the same configuration.

In Figure 12 we show an example of a configuration, that is not angle-rigid, however it cannot be classified as a *special not angle-rigid configuration* (Definition 3.5).

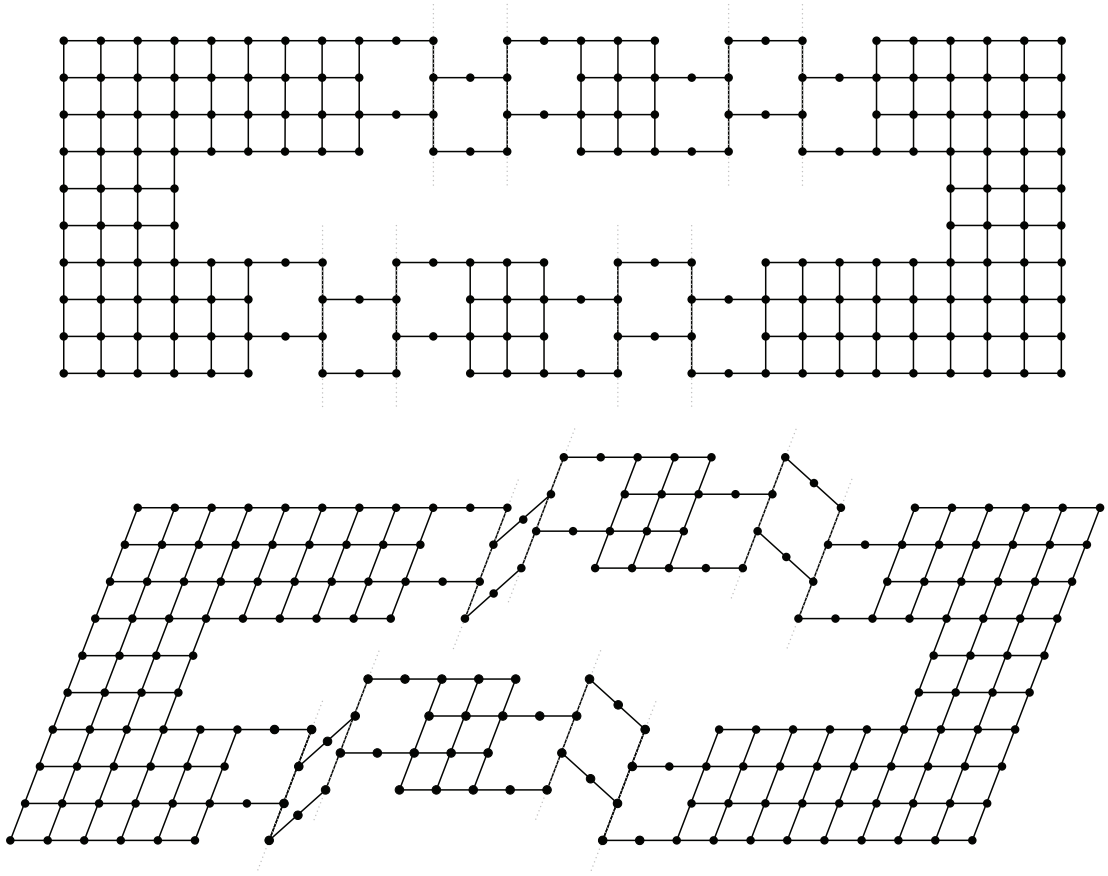

Figure 12: An example of a not angle-rigid configuration, that is not covered by this thesis.

Although the configuration in Figure 12 has rotational axes, we cannot take any one of them separately and rotate a part of the configuration around it without changing either bond lengths or bond angles. In order to deform the given configuration **and** keep both bond angles and lengths unchanged, we need to use all 8 axes simultaneously. Only then we can pop up the two middle pieces and slide both left and right bottom pieces towards each other. One important property of this configuration is, that the quadruples of axes above (a_1, a_2, a_3, a_4) and below (b_1, b_2, b_3, b_4) need to be shifted relative to each other, i.e. no two axes with the same index should coincide. Otherwise we would get a special not angle-rigid configuration by a book-flip.

3.3 Main results

Theorem 3.7 (Sufficient condition)**.** *Let $C \subset \mathbb{Z}^{2N}$ be connected and fulfill the following condition*

> *there exists an axis α such that, by removing from C all points on α having at most one neighbor off α, the resulting configuration is disconnected.* (3)

Then, C is a special not angle-rigid configuration.

Proof. Let C be connected and fulfill (3). Call $\widetilde{C}$ the configuration obtained by removing all points of C belonging to α and having at most one neighbor off α. Fix $C_1 \subset \widetilde{C}$ to be a connected component of $\widetilde{C}$ such that C_1 and $\widetilde{C}\backslash C_1$ are disconnected. Let the matrix $R(t) \in \mathbb{M}_{3,3}(\mathbb{R})$ represent a rotation around the axis α of amplitude t/θ, with $\theta < \pi/2$, and define the mapping $\varphi : [0,1] \times C \to \mathbb{R}^{3N}$ as

$$\varphi_t(x) = \begin{cases} R(t)x & \text{if } \ x \in C_1 \\ x & \text{if } \ x \in C\backslash C_1. \end{cases}$$

As C_1 and $\widetilde{C}\backslash C_1$ are disconnected, the map φ fulfills (1)-(2) when restricted to $\widetilde{C}$. We aim at showing that, by reintegrating the points in $C\backslash\widetilde{C}$ which were removed under condition (3), relations (1)-(2) still hold. We proceed sequentially, by adding one point at a time. To this aim, assume to be given a configuration $\widehat{C}$ with $\widetilde{C} \subset \widehat{C} \subset C$ such that (1)-(2) hold for $\widehat{C}$ but $\widehat{C} \neq C$. With no loss of generality we let $\alpha = \mathbb{Z} \times \{0\}$ and $(0,0) \in C\backslash\widehat{C}$. We now check that, by adding the point $(0,0)$ to $\widehat{C}$, properties (1)-(2) still hold true. To prove this, we need to check that the new bonds and angles that have to be included in the energy count by considering the extra point $(0,0)$ are still invariant under φ. Such new bonds and angles are determined by the neighbors and the triplets

$$N' = N(\widehat{C} \cup (0,0))\backslash N(\widehat{C}) \ \text{ and } \ T' = T(\widehat{C} \cup (0,0))\backslash T(\widehat{C}),$$

respectively.

As $(0,0) \notin \widetilde{C}$, we have that at least one point among $(0,-1)$ and $(0,1)$ does not belong to $\widehat{C}$. In case both do not belong to $\widehat{C}$, one has that the new bonds N' are necessarily aligned with α, namely

$$N' \subset \big\{((0,0),(1,0)),\, ((1,0),(0,0)),\, ((0,0),(-1,0)),\, ((-1,0),(0,0))\big\}. \tag{4}$$

As $\varphi_t(\cdot)$ is the identity on α, we have that $|\varphi_t(x_k) - \varphi_t(x_{k'})| = 1$ for all $(x_k, x_{k'}) \in N'$ and all $t \in [0,1]$ as well. As regards triplets, we have that

$$(x_k, x_{k'}, x_{k''}) \in T' \ \Rightarrow \ \text{at least two out of } x_k,\, x_{k'}, \text{ or } x_{k''} \text{ belong to } \alpha. \tag{5}$$

As $\varphi_t(\cdot)$ is the identity on α and either the identity or a rotation about α out of α, one has that $\theta(\varphi_t(x_k), \varphi_t(x_{k'}), \varphi_t(x_{k''})) = \theta(x_k, x_{k'}, x_{k''})$ for all $(x_k, x_{k'}, x_{k''}) \in T'$. In particular, properties (1)-(2) hold for $\widehat{C} \cup (0,0)$. Let us now consider the case $(1,0) \in \widehat{C}$ (the

remaining case $(-1,0) \in \widehat{C}$ can be treated analogously). Here, all bonds in N' are either aligned with α as in (4), or orthogonal to α. In both cases, as $\varphi_t(\cdot)$ is the identity on α and either the identity or a rotation about α out of α, one has that $|\varphi_t(x_k) - \varphi_t(x_{k'})| = 1$ for all $(x_k, x_{k'}) \in N'$ and all $t \in [0,1]$ as well. Those triplets in T' containing at least two points on α can be treated as before, see (5). We are just left to consider the triplets of the form

$$\big((0,0),(0,1),(0,2)\big),\ \big((0,0),(0,1),(1,1)\big),\ \big((0,0),(0,1),(-1,1)\big),$$

(and permutations $(x_k, x_{k'}, x_{k''}) \mapsto (x_{k''}, x_{k'}, x_k)$), if at all included in T'. Let us assume that $((0,0),(0,1),(1,1)) \in T'$. Then, $((0,1),(1,1)) \in N(C)$, implying that $(0,1)$ and $(1,1)$ belong to the same connected component of $\widetilde{C}$. In case $(0,1),\ (1,1) \in \widetilde{C} \backslash C_1$ we have that $\varphi_t(\cdot)$ is the identity on $(0,0)$, $(0,1)$, and $(1,1)$. In case $(0,1),\ (1,1) \in C_1$ we have that $\varphi_t(\cdot)$ is a rotation on $(0,0)$, $(0,1)$, and $(1,1)$. In both cases, $\varphi_t(\cdot)$ is an isometry on $(0,0)$, $(0,1)$, and $(1,1)$, implying that $\theta(\varphi_t(0,1), \varphi_t(0,1), \varphi_t(1,1)) = \theta((0,0),(0,1),(1,1))$. An analogous argument applies to $((0,0),(0,1),(0,2))$ and $((0,0),(0,1),(-1,1))$. We have hence proved that

$$\theta(\varphi_t(x_k), \varphi_t(x_{k'}), \varphi_t(x_{k''})) = \theta(x_k, x_{k'}, x_{k''}) \forall (x_k, x_{k'}, x_{k''}) \in T'.$$

Eventually, also in case $(1,0) \in \widehat{C}$, properties (1)-(2) hold true for $\widehat{C} \cup (0,0)$. We have hence proved that the configuration $\widehat{C} \cup (0,0) \subset C$ fulfills (1)-(2). In case $\widehat{C} \cup (0,0) \neq C$ we repeat the argument by adding a point from $C \backslash \widehat{C} \cup (0,0)$ to $\widehat{C} \cup (0,0)$, still preserving properties (1)-(2). Since the number of points in C is finite, this procedure eventually terminates, proving that C fulfills (1)-(2). As φ_t is not trivial for all $t \in (0,1]$, this proves that C is not angle-rigid. □

Theorem 3.8 (Necessary condition)**.** *Let $C \subset \mathbb{Z}^{2N}$ be connected. If C is a special not angle-rigid configuration, then there exists an axis α such that, by removing from C all points on α having at most one neighbor off α, the resulting configuration is disconnected.*

Proof. Let C be a special not angle-rigid configuration and let α and $\widetilde{C}$ be the axis and the subconfiguration from Definition 3.5. Without loss of generality we can assume that $\widetilde{C}$ is connected (if $\widetilde{C}$ is not connected, we redefine it to be one of its connected components). By changing coordinates, we can assume with no loss of generality that $\alpha = \mathbb{Z} \times \{0\}$.

For all $t \in (0,1]$ fixed, the mapping φ_t does not preserve the length of bonds connecting points in $\widetilde{C}$ and $C \backslash \widetilde{C}$ not belonging to α. This implies that $\widetilde{C}$ and $C \backslash \widetilde{C}$ are connected just through bonds along the axis α.

We would like to prove that condition (3) holds with this same axis α. In order to do this, we shall prove that by removing from C all points on α having at most one neighbor off α, the resulting configuration $\widehat{C}$ is disconnected.

Assume by contradiction that $\widehat{C}$ is connected. The two sets $\widetilde{C} \backslash \alpha$ and $(C \backslash \widetilde{C}) \backslash \alpha$ are nonempty and disjoint. Take a path connecting a point in $\widetilde{C} \backslash \alpha$ to a point in $(C \backslash \widetilde{C}) \backslash \alpha$.

As we have already proved that such path should contain bonds along α, with no loss of generality, by possibly redefining coordinates, we can find a path of the form

$$\{(0,1),(0,0),\ldots,(k,0),(k+1,0)(k+1,\pm 1)\} \subset \widehat{C}$$

where $k \geqslant 1$ is given, $(0,1) \in \widetilde{C}$ and $(k+1,1) \in C \backslash \widetilde{C}$ (or, with a completely analogous argument, $(k+1,-1) \in C \backslash \widetilde{C}$). As all points $(0,0),\ldots,(k,0)$ belong to $\widehat{C}$ after the removal described in (3), one has that all points $(0,\pm 1),\ldots,(k,\pm 1)$ belong to $\widehat{C}$ as well. As $(0,1) \in \widetilde{C}$ and $\widetilde{C}$ is connected we have that $(0,1),\ldots,(k,1)$ belong to $\widetilde{C}$. In particular, $(k,1) \in \widetilde{C}$ and $(k+1,1) \in C \backslash \widetilde{C}$. Hence, we have that

$$|\varphi_t(k,1) - \varphi_t(k+1,1)| = |R(t)(k,1) - (k+1,1)| > 1$$

contradicting (1). We conclude that $\widehat{C}$ is necessarily disconnected and (3) holds. □

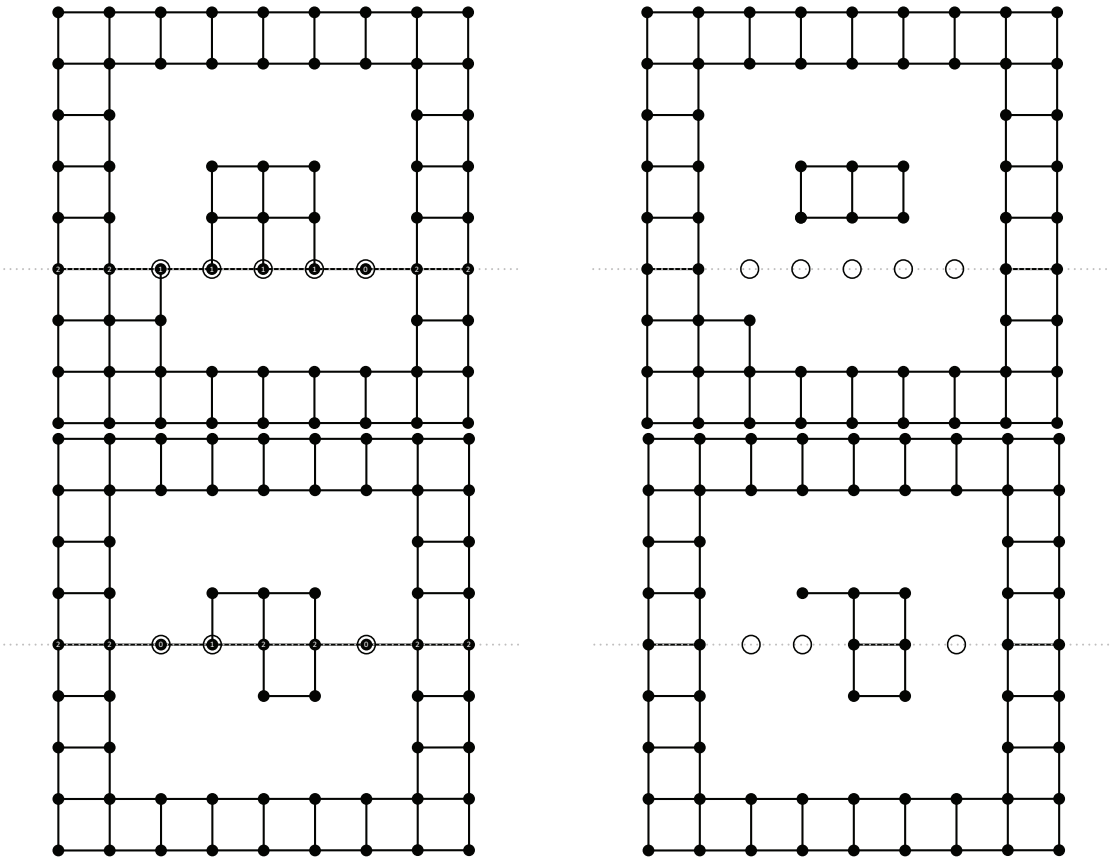

Figure 13: An illustration to the Theorems 3.7, 3.8 and the Corollary 3.9.

Corollary 3.9. *Let $C \subset \mathbb{Z}^{2N}$ be connected. C is a special not angle-rigid configuration if and only if there exists an axis α such that, by removing from C all points on α having at most one neighbor off α, the resulting configuration is disconnected.*

4 Numerical approach, simulations and assumptions

In the last part of the thesis we would like to take a numerical approach to our topic. The goal of the experiment below is to investigate, how probable it is, that some random connected configuration is a special not angle-rigid one.

4.1 Conditions and parameters

Firstly, we would like to introduce some conditions and parameters we would like to look at and work with.

The main restriction in our experiment is that we are going to look only at configurations constrained in a square of a certain side length n, which we will refer to as "square dimension". The square dimension n will be our first parameter. The second parameter is the percentage of the points, that are present in our configuration. For example: $n = 15$, percentage$= 75\%$ means, that we will randomly remove $15 \cdot 15 \cdot (1 - 0.75) = 56.25$ points. In order to get an integer number of points, we apply the ceiling function to the result: $\lceil 56.25 \rceil = 57$ points, which means, that the number of remaining points will be $15 \cdot 15 - 57 = 168$ points.

This of course means that for some fixed pair of parameters we will always get the same amount of points. The alternative approach of giving each individual point a probability of existence would increase the runtime of the code at least quadratically with each next square dimension. So, our workaround was that this fixed number would represent some kind of an average value of existing points of a substantially large amount of runs.

We have run our simulation in the software Wolfram Mathematica 11.0 on Intel Core i5-5200U and Intel Core i7-9700K.

4.2 Algorithm description

The algorithm, that we have used, can be described in the following way:

1. We create an empty matrix, which will in the end contain all the values, generated throughout all runs.

2. Then we define our square dimension range, that we would like to look at, e.g. from 7 to 16, which means, that the code will generate and analyze the 7×7 square configurations, then 8×8 and so on until we reach 16×16.

3. The next step is to create an empty vector, that will contain our desired values, that were generated for this specific square dimension.

4. Then we generate a full $n \times n$ square grid, i.e. a configuration that has all n^2 points. And we also, for the convenience purposes enumerate all points from left to right, from bottom to the top.

5. The next loop of our iteration is about the second parameter of percentage of the present points. Here we define a certain range of percentages, e.g. from 80 to 92, and then we immediately multiply this parameter with 0.01.

   We would like to notice here that it is of course possible to define our range more precisely, e.g. from 804 to 918, and multiply with 0.001, which will give us more data, but will run longer.

6. Then we create the counting variable, which will tell us, how many of our runs result in a desired configuration.

7. Now we need to actually create our configurations. We calculate how many points need to be removed: we apply the ceiling function to the product of the total possible number of points and (1-(percentage of remaining points))

8. Here we define how many runs we want to have for each pair of parameters. In our case we did 1000 runs.

9. Then we take the number of removed points and generate a list of that many random points from 1 to $n \times n$.

10. Next we delete the points from the full square grid.

11. Here we check whether the resulting configuration is still connected. If no - we return to the step 9. If yes - we continue.

12. Now we go through all columns from 1 to n, from left to right. For every point in the chosen column we check how many out-of-column neighbors this point has. If the column has at least one point that has either zero or one out-of-column neighbors (but not both at the same time!) — we increase the counter by 1 and go for the next run (step 9). If no column has such point — we continue.

13. Since no column gave us the desired result we check the rows in the same manner: from 1 to n, from bottom up. For every point in the chosen row we check how many out-of-row neighbors this point has. If the row has at least one point that has either zero or one out-of-row neighbors (but not both at the same time!), we increase the counter by 1 and go for the next run (step 9). If no row has such point we leave the counter unchanged, but we go to the step 9 anyway.

14. When we are done with all runs for the current pair of parameters, we add the counter value (of "successful runs") to the initial vector, and we continue from the step 6 with the next percentage value.

15. When we are done looking at all percentages of the given square dimension we add the resulting vector to the final matrix and we continue from the step 3 with the next square dimension.

16. When we are done with all square dimensions we get our final matrix full of values from all runs. Done.

Using the pseudocode this algorithm looks in the following way:

```
countermatrix = {};
for n = 7 to 20 do
    ran = n · n;
    contervector = {};
    g = full grid with enumerated points from left bottom to right top;
    for prob = 70 to 90 do
        probability = prob · 0.01;
        counter = 0;
        no = ⌈ran · (1 − probability)⌉;
        for iteration = 1 to 1000 do
            begin;
            list = no-many random points in the range from 1 to n · n;
            g2 = full grid with the points from list removed;
            savelist = list, as a backup;
            if g2 has more than 1 connected component then
                goto begin;
            end
            for column = 1 to n do
                list2 = savelist;
                for vertex = column to column + n · n − n by n do
                    if vertex has both left and right neighbor then
                        continue;
                    else
                        add vertex to the list2;
                    end
                end
                list1 = list2, but without duplicates, should they occur;
                g4 = full grid with the points from list1 removed;
                if number of connected components in g2 and g4 is different then
                    counter++;
                end
            end
            for row = 1 to n do
                list2 = savelist;
                for vertex = row · n − n + 1 to row · n do
                    if vertex has both above and below neighbor then
                        continue;
                    else
                        add vertex to the list2;
                    end
                end
                list1 = list2, but without duplicates, should they occur;
                g4 = full grid with the points from list1 removed;
                if number of connected components in g2 and g4 is different then
                    counter++;
                end
            end
        end
        add counter to countervector;
    end
    add countervector to countermatrix;
end
output countermatrix;
```

4.3 Results, visualization of the data and further assumptions

In the plots in Figures 14, 15 we can see the distribution of values, that were computed during our code runs. We look first at the first of the two Figures (14). The vertical axis represents the number of runs, that delivered us a special not angle-rigid configuration. The bottom axis represents the square dimensions from 6 to 40. Finally, the right axis represents the percentage of remaining points in the square grid from 70% to 99%.

The plot in Figure 15 displays the same distribution, but rotated, such that the bottom axis represents the percentage of remaining points in the square grid from 70% to 99%, and the right axis shows the square dimensions. The vertical axis still shows the number of "successful" runs.

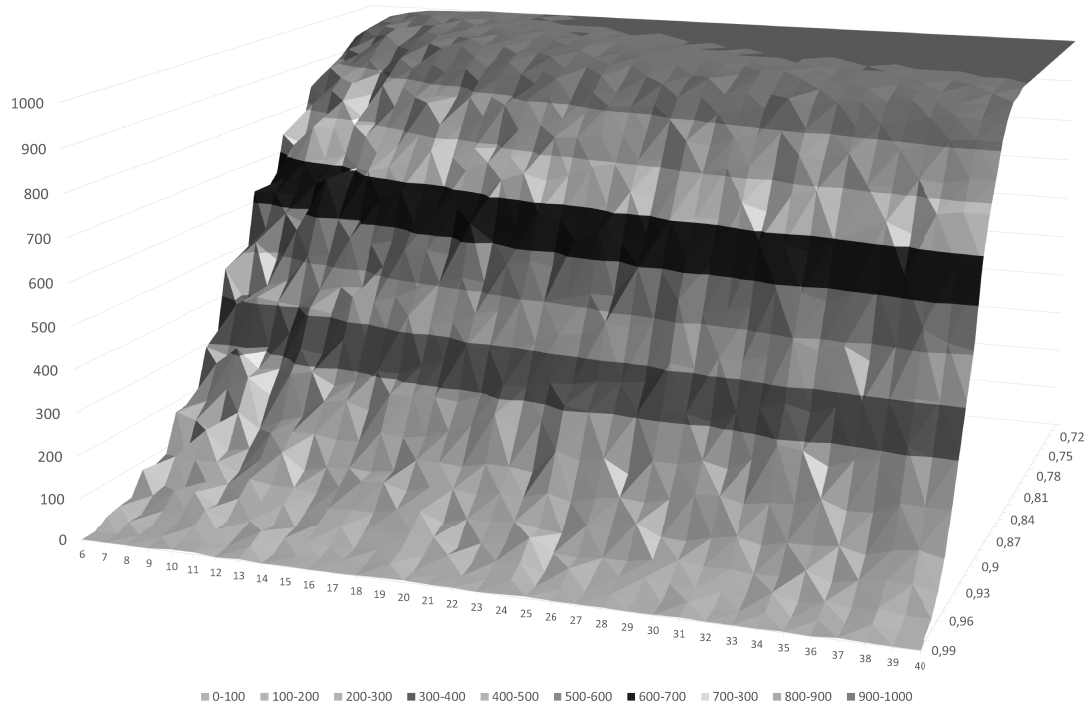

Figure 14: 3d plot of the runs.

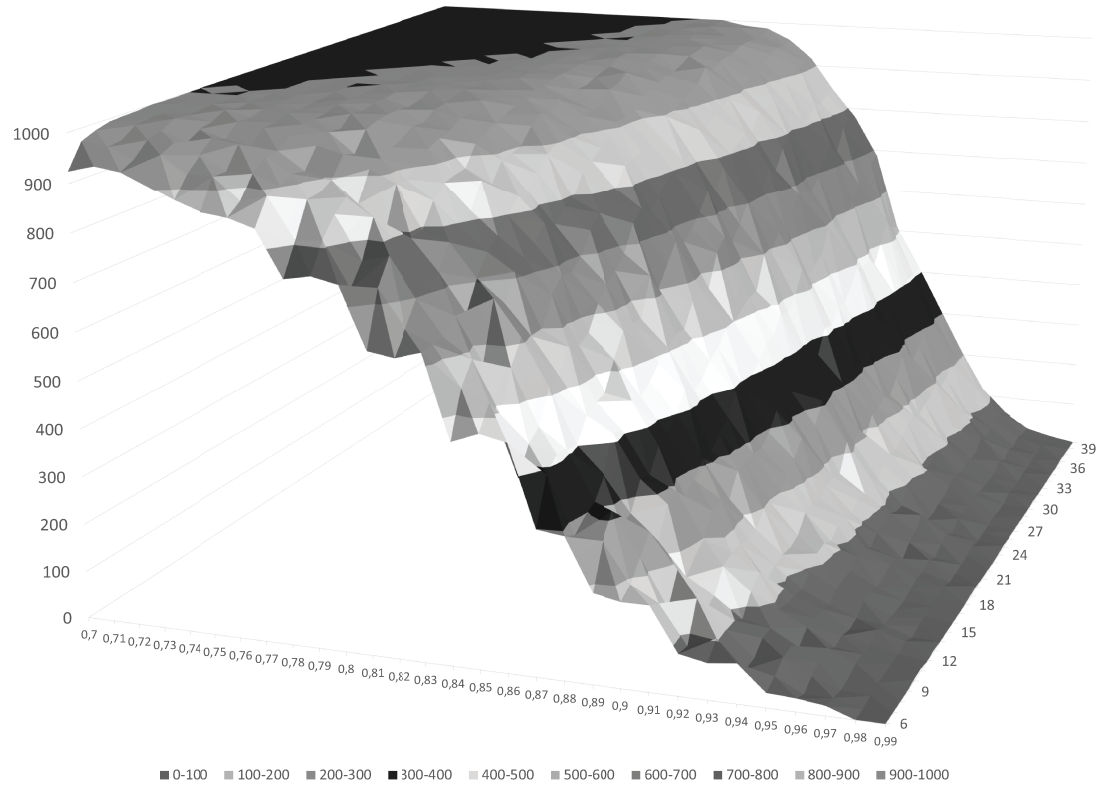

Figure 15: 3d plot of the runs.

The plot in Figure 16 represents the individual 2–dimensional slices of the plots above. For clarity, we will refer to the first of the two distributions above (Figures 14, 15). The slices were made at each square dimension mark, parallel to the right axis with percentages. As we may see at first, for low dimensions the distributions look very jagged and they start to drop down from 1000 to 0 pretty much immediately, but slowly. On the other hand, the higher dimensions have more smooth distributions of values, and they start to drop down much later but more rapidly.

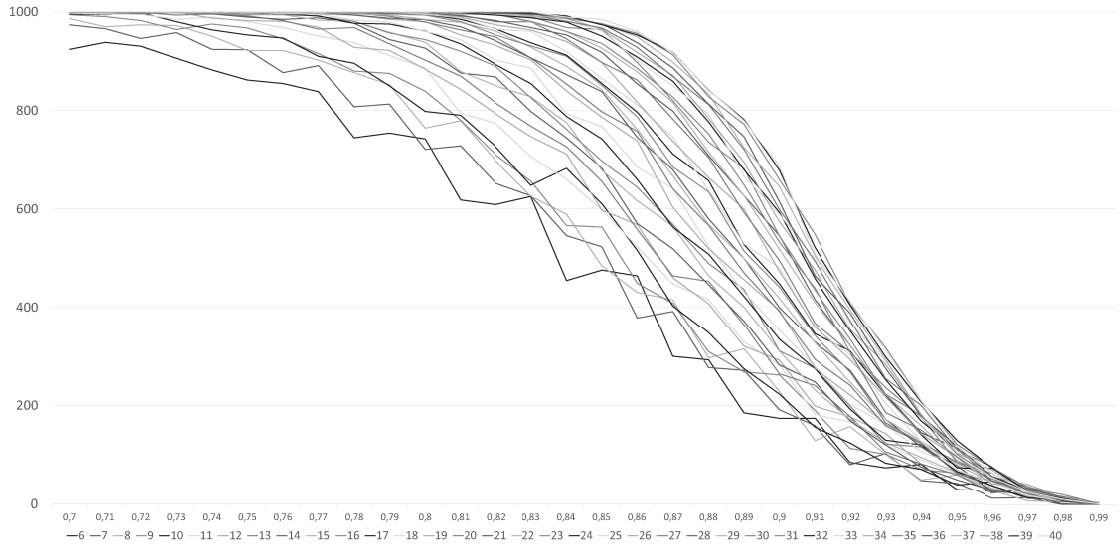

Figure 16: 2d plot of the runs.

We conjecture, that if the code would be run on the systems with much higher performances, which would allow us to go to much higher square dimensions, we might see even smoother distributions with even more rapid drops from maximal possible number of "successful" runs to 0, and it even might "converge" to an "immediate" drop.

For the interested reader the Wolfram Mathematica sourcecode may be found in the Appendix A.

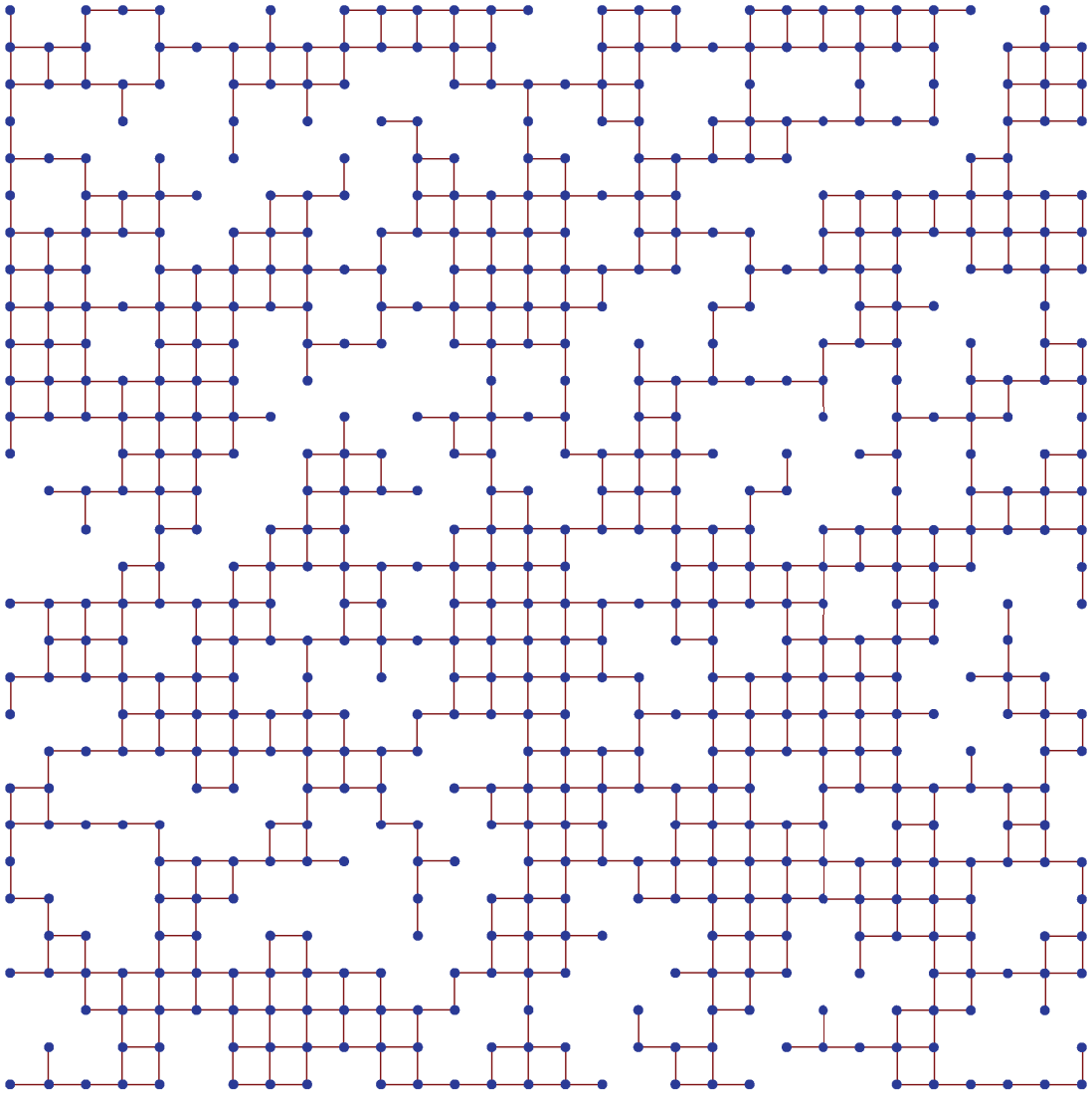

Figure 17: An example of a configuration, generated by Wolfram Mathematica with parameters n=30, p=0.75. This configuration is a special not angle-rigid one: the rotational axes could be placed at the rows 1, 2, 8, 9, 12, 17, 25, 28 (counting from bottom up) or at the columns 11, 12, 20, 21, 25, 28, 29 (counting left to right).

A Wolfram Mathematica sourcecode

```
countermatrix = {};
For[n = 7, n <= 20, n++,
  ran = n*n;
  countervector = {};
  g = SetProperty[GridGraph[{n, n}], VertexCoordinates ->
  Flatten[Array[{#2, #1} &, {n, n}], 1]];
  For[prob = 70, prob <= 90, prob++,
    probability = prob*0.01;
    counter = 0;
    no = Ceiling[ran*(1 - probability)];
    For[iteration = 1, iteration <= 1000, iteration++,
      Label[begin];
      list = RandomSample[Range[ran], no];
      g2 = VertexDelete[g, list];
      savelist = list;
      If[Length[ConnectedComponents[g2]] > 1,
        Goto[begin]
      ]
      Catch[
        For[column = 1, column <= n, column++,
          list2 = savelist;
          For[vertex = column, vertex <= column + n*n - n, vertex += n,
            If[
              And[
                MemberQ[VertexList[g2], vertex],
                MemberQ[VertexList[g2], vertex - 1],
                MemberQ[VertexList[g2], vertex + 1]
              ],
              Continue[],
              AppendTo[list2, vertex]
            ]
          ];
          list1 = DeleteDuplicates[list2];
          g4 = VertexDelete[g, list1];
          If[Length[ConnectedComponents[g4]] != Length[ConnectedComponents[g2]],
            Throw[counter++]
          ]
        ]
        For[row = 1, row <= n, row++,
          list2 = savelist;
          For[vertex = row*n - n + 1, vertex <= row*n, vertex++,
            If[
              And[
                MemberQ[VertexList[g2], vertex],
                MemberQ[VertexList[g2], vertex - n],
                MemberQ[VertexList[g2], vertex + n]
              ],
```

```
                Continue[],
                AppendTo[list2, vertex]
              ]
            ];
            list1 = DeleteDuplicates[list2];
            g4 = VertexDelete[g, list1];
            If[Length[ConnectedComponents[g4]] != Length[ConnectedComponents[g2]],
              Throw[counter++]
            ]
          ]
        ]
      ]
      AppendTo[countervector, counter]
    ]
    AppendTo[countermatrix, countervector]
]
countermatrix
```

As a final remark to the code above: we do not claim that this particular code is optimized to give the result in the fastest time possible.

References

[1] *J. Capco, M. Gallet, G. Grasegger, C. Koutschan, N. Lubbes, J. Schicho*, The number of realizations of a Laman graph, SIAM J. Appl. Algebra Geom. 2, No. 1, 94–125 (2018; Zbl 06885643).

[2] *B. J. Cox, J. M. Hill*, Exact and approximate geometric parameters for carbon nanotubes incorporating curvature, Carbon, V. 45, I. 7, 2007, P. 1453–1462, ISSN 0008-6223.

[3] *I. Z. Emiris, E. P. Tsigaridas, A. Varvitsiotis*, Distance geometry. Theory, methods, and applications. New York, NY: Springer. 23–45 (2013; Zbl 1269.05077).

[4] *J. L. Gross, J. Yellen, M. Anderson*, Graph theory and its applications. 3rd edition. Boca Raton, FL: CRC Press (2019; Zbl 1404.05001).

[5] *G. C. Hamrick, C. Radin*, The symmetry of ground states under perturbation. J. Stat. Phys. 21, 601–607 (1979).

[6] *G. Laman*, On graphs and rigidity of plane skeletal structures, J. Eng. Math. 4, 331–340 (1970; Zbl 0213.51903).

[7] *E. Mainini, P. Piovano, U. Stefanelli*, Finite crystallization in the square lattice, Nonlinearity 27, No. 4, 717–737 (2014; Zbl 1292.82043).

[8] *R. Olfati-Saber, R. Murray*, Graph rigidity and distributed formation stabilization of multi-vehicle systems. Proceedings of the IEEE Conference on Decision and Control. 3. 2965–2971 vol.3. 10.1109/CDC.2002.1184307, December 2002.

[9] *S. Pirzada*, Applications of graph theory. Proc. Appl. Math. Mech., 7: 2070013-2070013, 2007.

[10] *J. Rollin, L. Schlipf, A. Schulz*, Recognizing Planar Laman Graphs. ESA 2019: 79:1–79:12.

[11] *U. Stefanelli*, Stable carbon configurations, Boll. Unione Mat. Ital. 10, No. 3, 335–354 (2017; Zbl 1380.82052).

[12] *V. Yegnanarayanan*. Graph theory to pure mathematics: Some illustrative examples. Resonance. 10. 50-59. 2005 10.1007/BF02835892.