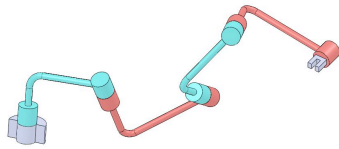


Computational Geometry algorithms for Robot Manipulators, with applications

ICRA 2010 Tutorial
Friday May 7th, 2-5:30 - Anchorage, Alaska



ORGANIZER: Ileana Streinu

Department of Computer Science
Department of Mathematics
Smith College and University of Massachusetts Amherst
Northampton, MA 01063

Contact information

[HOME](#)

[ABSTRACT](#)

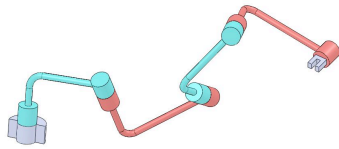
[OBJECTIVES](#)

[SCHEDULE](#)

[REFERENCES](#)

Computational Geometry algorithms for Robot Manipulators, with applications

ICRA 2010 Tutorial
Friday May 7th, 2-5:30 - Anchorage, Alaska



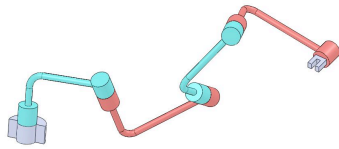
ABSTRACT

The purpose of this tutorial is to expose the Robotics community to a series of theoretical results and efficient algorithms for robotic manipulators based on techniques from Computational Geometry and Rigidity Theory. These will include the Carpenter's Rule algorithm for non-colliding motion planning of planar robot arms, the recent mathematical solution to the Reach Problem for manipulators with revolute joints, as well as applications to origami folding and flexibility analysis of biological macromolecules from a robotics perspective.

[HOME](#)[ABSTRACT](#)[OBJECTIVES](#)[SCHEDULE](#)[REFERENCES](#)

Computational Geometry algorithms for Robot Manipulators, with applications

ICRA 2010 Tutorial
Friday May 7th, 2-5:30 - Anchorage, Alaska



MOTIVATION and OBJECTIVES

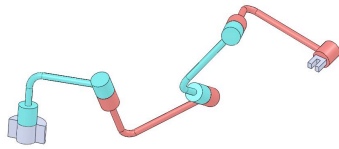
Theoretical and algorithmic advances on classical Robotics problems recently developed in Computational Geometry are virtually unknown to the Robotics community.

The objective of this tutorial is to make these new algorithmic tools accessible to the robotics audience and to demonstrate their usefulness in the context of two application areas: modeling flexibility of large biomolecules and origami folding. The presentation, while rooted in classical mathematics, will be very accessible and rely on a variety of demos and physical props.

[HOME](#)[ABSTRACT](#)[OBJECTIVES](#)[SCHEDULE](#)[REFERENCES](#)

Computational Geometry algorithms for Robot Manipulators, with applications

ICRA 2010 Tutorial
Friday May 7th, 2-5:30 - Anchorage, Alaska



TENTATIVE SCHEDULE and AGENDA

Friday May 7
2:00-5:30

Three 50-minute lectures separated by two coffee breaks will present the mathematical background, algorithms and applications, and allow for discussion time.

[HOME](#)

[ABSTRACT](#)

[OBJECTIVES](#)

[SCHEDULE](#)

[REFERENCES](#)

PRESENTERS: The lectures will be given by the organizer.

INTENDED AUDIENCE: members of the motion planning, assembly planning, nano-robotics and manipulation communities.

LIST OF TOPICS:

The **Carpenter's Rule Problem**: non-colliding motion planning of planar robot arms and loops

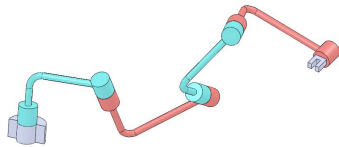
The **Reach Problem**: extremal configurations of robotic manipulators with revolute joints

Spherical linkages and origami folding

Flexibility analysis of Biological Macromolecules based on Robot-Kinematics-Rigidity Models

Computational Geometry algorithms for Robot Manipulators, with applications

ICRA 2010 Tutorial
Friday May 7th, 2-5:30 - Anchorage, Alaska

[HOME](#)[ABSTRACT](#)[OBJECTIVES](#)[SCHEDULE](#)[REFERENCES](#)

REFERENCES

[1] How far can you reach?

Ciprian S. Borcea and Ileana Streinu

in Proc. ACM-SIAM Symposium on Discrete Algorithms (SODA10), 928-937, Austin, Texas, 2010.

[2] Flattening single-vertex origami: the non-expansive case

Gaiane Panina and Ileana Streinu

in Proc. 25th Annual ACM Symposium on Computational Geometry, 316-323, 2009.

[3] Extremal Configurations of Manipulators with Revolute Joints

Ciprian S. Borcea and Ileana Streinu

in Reconfigurable Mechanisms and Robots, King's College, London, UK, 279-284

Proc. ASME/IFTOMM International Conference (ReMAR'09), Jian S. Dai, Matteo Zoppi and Xianwen Kong (eds.), Genova, 2009.

[4] How Hydrogen Bonds Affect Protein Rigidity

Naomi Fox, Filip Jagodzinski, Jeanne Hardy, Ileana Streinu

in 23rd Symposium of the Protein Society, Proteins in Motion. July 25-29, 2009 Boston Marriott Copley Place, Boston, Massachusetts, 2009.

[5] A Web-Based Tool for Rigidity Analysis of Proteins

Naomi Fox, Filip Jagodzinski, Yang Li, Ileana Streinu

in Biotechnology and Bioinformatics Symposium (BIOT 2009), Lincoln, Nebraska, 2009.

[6] A Software Tool for Surveying the Rigidity Properties of Protein Families

Filip Jagodzinski, Naomi Fox, Diana Jaunzeikare, Ileana Streinu

in Biotechnology and Bioinformatics Symposium (BIOT 2009). October 9-10, 2009 Lincoln, Nebraska, 2009.

[7] Analyzing rigidity with pebble games

Audrey Lee, Ileana Streinu, Louis Theran

in SOCG '08: Proceedings of the twenty-fourth annual symposium on Computational geometry, 226-227, College Park, MD, USA, 2008.

[8] Pseudo-Triangulations, Rigidity and Motion Planning

Ileana Streinu

Discrete and Computational Geometry, 34(4), 587-635, 2005.

[9] Single-Vertex Origami and Spherical Expansive Motions

Ileana Streinu and Walter Whiteley

in Proc. Japan Conf. Discrete and Computational Geometry (JCDCG 2004), 161-173, 2005.

[10] A Combinatorial Approach to Planar Non-Colliding Robot Arm Motion Planning

Ileana Streinu

Proc. 41st IEEE Annual Symposium on Foundations of Computer Science (FOCS), 443-453, 2000.

How far can you reach?

Ciprian Borcea¹ and Ileana Streinu²

Abstract

The problem of computing the maximum reach configurations of a 3D revolute-jointed manipulator is a long-standing open problem in robotics. In this paper we present an optimal algorithmic solution for orthogonal polygonal chains. This appears as a special case of a larger family, fully characterized here by a technical condition. Until now, in spite of the practical importance of the problem, only numerical optimization heuristics were available, with no guarantee of obtaining the global maximum. In fact, the problem was not even known to be computationally solvable, and in practice, the numerical heuristics were applicable only to small problem sizes.

We present elementary and efficient (mostly linear) algorithms for four fundamental problems: (1) finding the maximum reach value, (2) finding a maximum reach configuration (or enumerating all of them), (3) folding a given chain to a given maximum position, and (4) folding a chain in a way that changes the endpoint distance function monotonically. The algorithms rely on our recent theoretical results characterizing combinatorially the maximum of panel-and-hinge chains. They allow us to reduce the first problem to finding a shortest path between two vertices in an associated simple triangulated polygon, and the last problem to a simple version of the planar carpenter’s rule problem.

1 Introduction

The *revolute-jointed robot arms* considered in this paper are spatial polygonal chains with fixed edge lengths and fixed angles between consecutive edges. The possible motions of the chain are rotations about the interior edges. These edges are called *revolute joints* or, when viewed as entire lines, *hinges*. As the robot arm moves between possible configurations, the distance between the two endpoints of the polygon (the *endpoint distance function*) takes on a continuum of values, of which the

maximum one makes the focus of this paper. We remark that our chains are allowed to self-intersect during the motion. The following is a basic, long-standing open question in robotics, where it appears in connection with the much studied workspace computation problem [19, 3]:

The Maximum Reach Problem. *Given a 3D revolute-jointed robot arm, compute the maximum value of the endpoint distance function and a corresponding configuration.*

The main result of this paper is a *surprisingly simple, optimal, linear time algorithms for finding a maximum reach configuration for orthogonal chains*, and other classes (fully characterized by a technical condition). We show how to find the maximum value of the endpoint distance function, one of the (possibly exponentially many) maximum reach configurations, and how to fold to one of these configurations in a linear number of steps.

Although research on this problem spans over 40 years (see, e.g. [10, 8, 6, 1]), theoretical results contributed little to computational advances. The typical approach was based on formulating it as an optimization problem. In practice, for small-sized instances, it was solved with numerical approximation methods. Even very basic questions, such as whether a candidate solution can be *verified* in polynomial time, or whether the maximum is at all *computable* by a discrete algorithm, remained open. Indeed, in order to find the maximum, one has to look for it in a high dimensional space, described by the algebraic equations arising from the length and angle chain constraints. No *a priori* discrete underlying structure was known, to guide the search. For instance, if a gradient-based method landed in a potential well (local maximum), there was no theoretical criterion to distinguish a local maximum from the global one.

Papers studying the maximum reach of revolute jointed robotic manipulators have appeared in the robotics literature since at least 1969 [10], and gained momentum in the early 1980’s, when a general, necessary condition satisfied by *all critical points* of the endpoint distance function was identified [8, 13, 19, 16]. An approximation method received in 1985 the ACM

¹Department of Mathematics, Rider University, Lawrenceville, NJ 08648. email: borcea@rider.edu

²Department of Computer Science, Smith College, Northampton, MA 01063, USA. istreinu@smith.edu. streinu@cs.smith.edu, <http://cs.smith.edu/~streinu>.

³Research supported by a DARPA “23 Mathematical Challenges” grant, under “Algorithmic Origami and Biology”.

Distinguished Dissertation Award [12]. Recent activity focuses on effectively finding or approximating the workspace boundary for *specific* mechanical manipulators with few joints [2, 1]. A strong impetus comes from the applications of robotics methods in Molecular Biology (see e.g. [5]).

Our Results. We present combinatorial algorithms for the following four fundamental problems on robot arm reachability, applied to orthogonal chains:

1. **Maximum Reach Value:** compute the maximum value of the endpoint distance function.
2. **Maximum Reach Configuration:** compute one (or all, when the number is finite) of the configurations that achieve the global maximum of the endpoint distance.
3. **Motion Planning:** given an arbitrary configuration of the chain, reconfigure it to a maximum reach position: i.e. compute a trajectory in configuration space ending at a specific maximum reach configuration.
4. **Optimized Motion Planning:** given a flat configuration, reconfigure it in such a way that the distance function increases towards the maximum throughout the motion.

The class of chains for which these algorithms work is actually larger, and fully characterized by a technical condition, related to the triangle inequality on the sphere. This will be described in Section 5. In particular, this holds for chains with all angles equal to α , for $\alpha \geq \frac{\pi}{3}$. However, in order to avoid cluttering the presentation with technicalities, we'll focus on the orthogonal case.

The main effort goes into solving problem (1). We do so with a linear time algorithm for computing not just the maximum reach value, but also the fold points and the fold pattern of the flat pieces of the corresponding configuration. After that, in linear time we compute the *angles* by which the panels will be rotated at the fold points. A maximum reach *configuration* is then computed in linear time by standard forward kinematics calculations. Similarly, problem (3) can be solved by designing a trajectory that sequentially rotates the panels by the appropriate angles. The solution to problem (4) relies on expansive motions, and will be briefly sketched at the very end. To stay within our goal of designing *combinatorial* algorithms whose complexity can be described in terms on n , we use the $O(n^3)$ -events algorithm [17] for the planar Carpenter's Rule Problem.

Overview of the paper. We give the necessary definitions in Section 2, and state our previous theoretical results in Section 3. Along the way, we describe a structure theorem for maximum configurations, based on so-called rope segments and fold points, and present the equivalent shortest-path formulation which is more amenable to algorithmic treatment. The main algorithm appears in Section 4. It computes not just the length of the maximum reach, but also identifies, from a certain flat configuration, the structural elements of a maximal one. In Section 5, we describe the algorithm for computing a maximum reach configuration, which allows us to identify the technical condition which makes this process possible: the single-vertex origami folding. We now prove that the algorithm works correctly if and only if this condition is satisfied. In Section 7, we further refine the folding process to work in a manner that increases the endpoint distance monotonically. We conclude with some open problems.

2 Definitions

Revolute-jointed chains. A *robot arm* with n revolute joints (hinges) is given by a polygonal chain $p = \{p_0, p_1, \dots, p_{n+2}\}$ in $3D$, with fixed edge lengths and fixed angles between consecutive edges. The hinges correspond to the internal edges $e_i = (i, i+1)$, $i = 1, \dots, n$. The two points $s = p_0$ and $t = p_{n+2}$ are referred to as the *endpoints* of the chain, with s being the *start* or *origin*, and t the *terminus* or *end point*. Another way to look at such a chain is as follows: the fixed angle constraint turns all triplets of vertices $p_i p_{i+1} p_{i+2}$ into rigid triangles, since the length of the edge $p_i p_{i+2}$ is implied by the other two and by the angle between them. The plane of the triangle is called a *panel*, and consecutive panels $p_i p_{i+1} p_{i+2}$ and $p_{i+1} p_{i+2} p_{i+3}$ are joined by the hinge e_{i+1} running through $p_{i+1} p_{i+2}$. A *reminder*: a hinge should be conceived as an entire line, not just a line segment.

Panel-and-hinge chains. More generally, a *panel-and-hinge chain* is a sequence of panels connected by hinges. A *panel* is a plane, and a *hinge* is a line, rigidly attached to it. In a chain, all panels have exactly two hinges, except for the two extreme ones which contain just one hinge each. Two consecutive panels are free to rotate, one relative to the other, around their common hinge. An *origin* or *start point* is fixed on the first panel, and an *end-point* or *terminus* is marked on the last.

Panel-and-hinge chains allow for the case of parallel consecutive hinges or several consecutive hinges incident in the same point, but a *generic* chain won't have such degeneracies. If we connect the start point to a point

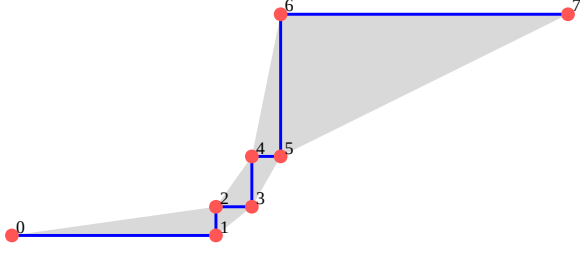


Figure 1: Rigid triangles in the panel-and-hinge chain associated to the revolute-jointed chain of Fig. 3(a).

on the first hinge by a line segment (an edge) and the terminus point to a point on the last hinge by another edge, we retrieve a revolute-jointed chain presentation with internal edges given by the segments on the hinges between two crossing points. The panels can be again conceived as triangles. See Fig. 1.

Endpoint axis and segment. The *line* (resp. line segment) joining the start and terminus is called the *endpoint axis* (resp. endpoint segment) of the chain.

Configuration space. The set of all possible spatial positions of the vertices which satisfy the edge length and angle constraints of a revolute-jointed chain (resp. panel-and-hinge), *up to rigid motions*, forms the *configuration space* of the chain. **We allow our chains to self-intersect.** The configuration space is naturally isomorphic to the n -dimensional torus $(S^1)^n$, for all the types of chains defined above. For panel-and-hinge chains, it can be parametrized by the dihedral angles between consecutive panels.

We emphasize that we address primarily the **generic** case in each class of chains. That means working on the complement of a proper algebraic subvariety of the parameter space of that class. Nevertheless, once a pattern is recognized, it is not difficult to see which aspects persist for the “non-generic” limit locus.

Flat configurations. When all the panels are coplanar, we say that the panel-and-hinge structure is in a *flat* configuration or simply *flat*. If the panels arise as triangles from a revolute-jointed polygonal chain, a special *standard* configuration is distinguished: when consecutive triangle do not overlap. Fig. 3 illustrates the standard *flat configuration* of an *orthogonal* chain (with equal right angles) having 5 hinges. Its panel-and-hinge representation from Fig. 1 illustrates the local non-overlapping property. More generally, it is easy to show that there is *no global overlap* for all chains with equal obtuse angles.

Endpoint (squared) distance. The *endpoint distance function* assigns a real non-negative value (the distance between the endpoints p_0 and p_{n+2}) to each spatial configuration of the chain. In fact, the squared distance function is more convenient for computations. The endpoint distance varies between two extreme values, the *global* minimum and maximum, with the possibility of various other local minima or maxima.

3 Theoretical background

The insufficient theoretical understanding of maximum configurations seems to be partially responsible for the lack of discrete (non-numerical) algorithms. A *necessary* condition for extremal non-zero⁵ configurations was recognized and proven in several papers [8, 13, 19, 16]. In the words of [16], this necessary condition says: “*the line of sight from the base-point to the hand must intersect all turning axes*”⁶, where the base-point may be chosen arbitrarily, the end-point is called “hand” and the “turning axes” are what we call hinges. However, all *critical points* with non-zero value for the squared distance function between the extreme points (not just the maxima) satisfy this condition.

Structure of critical configurations. As an immediate consequence of this condition, we obtain simple structural properties of panel-and-hinge chains in critical non-zero configurations. Two consecutive hinges may be met by the endpoint axis either in two distinct points (in which case, the axis lies in the plane, or panel, spanned by the two hinges), or in their intersection point; in this last case, the endpoint axis may not lie in the plane of the two hinges. If several consecutive hinges are met at distinct points, they (and consequently the panels they span) must be coplanar. Thus, the chain is subdivided into flat pieces (made by several consecutive coplanar panels cut by the endpoint axis), and connector panels. A connector panel is not coplanar with the endpoint axis. Instead, it meets the endpoint axis at the intersection of its two incident hinges. These special points, where the endpoint axis meets two (or more) concurrent hinges simultaneously, are called *fold points*.

In summary: the endpoint axis cuts across the hinges in the flat regions, and goes simultaneously through two consecutive hinges at fold points. See 8(b) for an example with two fold points; the endpoint axis meets three hinges in the middle region. *We will make*

⁵Configurations where the endpoint distance function is zero are also critical points, but they are not isolated for $n \geq 4$.

⁶This incidence of the origin-to-terminus line with the hinges is understood projectively, that is, it includes the possibility of parallelism.

substantial use of this property in the description of our algorithms.

Recently, in [4], we have extended this condition to a combinatorial, necessary and sufficient characterization of the Maximum Reach configurations. Because it helps to compare what is specific to the special case addressed in this paper, we state it in its full generality, for *body-and-hinge chains*. This is the most general class of serial robotic manipulators with revolute joints, defined as a collection of rigid *bodies* connected serially by hinges. The difference from panel-and-hinge chains is that the two hinges attached to each body (except the first and the last, which have only one hinge) need not be coplanar. As before, we mark two points, a *base* or *start point* on the first body, and an *end point*, or *terminus* on the last body, and ask for the maximum distance between them. The *natural order* of the hinges is $1, 2, 3, \dots$ as they appear on the chain. See Fig. 2. We have the following *complete theoretical characterization* (valid in arbitrary dimension):

THEOREM 3.1. [4] (Global Maximum) *A body-and-hinge chain is in a global maximum configuration if and only if the oriented segment from the origin s to the terminus t intersects all hinges in their natural order.*

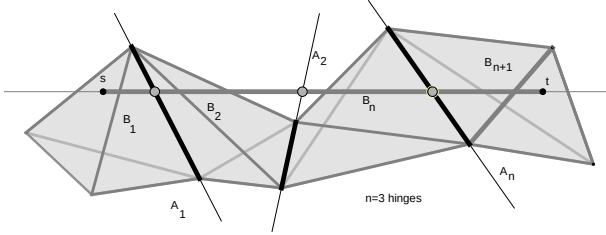


Figure 2: A body-and-hinge chain in R^3 with $n = 3$ hinges, 4 bodies (visualized as tetrahedra) and two marked points s and t on the end-bodies. In a maximum reach position, the axes meet the *oriented segment* st in the natural order.

Indeed, given a configuration satisfying this condition, we will mark in red the line segment from s to t , and think of its pieces, between the intersection points with the hinges, as being rigidly attached to the corresponding bodies. In any other configuration of the chain, the red path appears as a polygonal chain in 3D; hence the endpoint distance will be shorter than the length of the red path. The necessity of the condition is obtained from a characterization of the global maximum as a global minimum of another function:

THEOREM 3.2. [4] (Global Maximum as a Global Minimum) *The global maximum of the endpoint distance function coincides with the length of the shortest path from s to t which meets all hinges in their natural order.*

See Fig. 3 for an example of a flat configuration which is, and one which is not a maximum, as witnessed by the pattern of intersection of the hinges with the endpoint axis. Notice that flat configurations are automatically critical points of the endpoint distance function, since the endpoint axis and all hinges are coplanar, and therefore projectively incident. Later on, Fig. 8 illustrates a flat, non-maximal configuration and a corresponding global maximum.

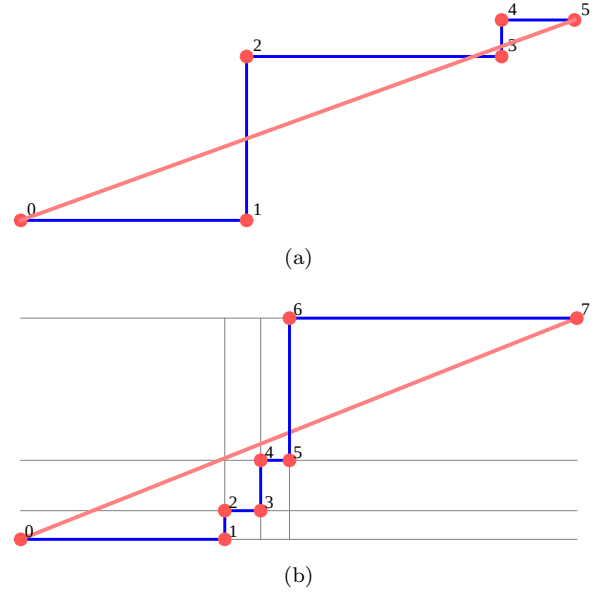


Figure 3: Illustration of Theorem 3.1. (a) This flat orthogonal chain is in its global maximum position, since the segment from the start to the terminus crosses the hinges in the natural order. (b) The hinges (in light gray) are crossed in a different order. The maximum reach requires a non-flat configuration in this case.

In this paper, we rely on these properties to devise the algorithms. The proofs of these Theorems have appeared in [4].

Notice that Theorem 3.1 immediately yields a simple linear time *verification algorithm* for the Maximum Reach. By contrast, the "classical" necessary condition of [8, 13, 19, 16] leads only to a verification procedure for being a *critical point*, not necessarily an maximum. The number of critical points of the endpoint distance function could be exponentially large, and - to the best of our knowledge - there is no known procedure that can

isolate from them the maxima, based on this information alone. Theorem 3.2 is also an essential ingredient in our algorithm for Maximum Reach, since it allows us to reduce its calculation to a constrained shortest path problem.

As we said, critical configurations of panel-and-hinge chains are subdivided into flat pieces connected at *fold points*. When a panel-and-hinge chain is folded, the angles induced by the two incident hinges at a fold point and the constrained shortest path between the endpoints will satisfy a simple condition related to the triangle inequality on the sphere (this is a crucial condition; we'll say more about it later in the paper). At each fold point, the incident panels can be (generically) folded in two distinct ways (the applicable concept of genericity includes most of the polygonal chains). We obtain:

THEOREM 3.3. [4] (Number of Extremal Configurations) *Generically, the number of distinct configurations of panel-and-hinge chains attaining the maximum reach is 2^f , where f is the number of fold points. All maxima are global.*

By contrast, for body-and-hinge chains, the global maximum is generically achieved by a unique configuration, and there may be exponentially many local maxima.

This theorem clarifies our goals: we will aim at computing the maximum reach value (which is unique) but not *the* maximum reach configuration (which is not). We will settle to folding the chain to *one* of the maximum reach positions, characterized by a certain pattern of orientations at fold points. The theorem also explains the observed behavior of gradient-based numerical methods, since there are no local (non-global) maxima (this is valid for all polygonal chains with fixed edge lengths and angles, not just the orthogonal ones).

4 Finding the maximum reach

We are ready now to describe the main algorithm. It finds the *value* of the maximum reach and computes additional information which will be used, in the next section, to compute one of the (possibly exponentially many) configurations in which the maximum can be attained. The proof of correctness, for orthogonal chains, is also addressed in the next section.

Recall the structural decomposition of a chain in a critical, and in particular in a maximum reach position described in the beginning of Section 3: it consists in flat regions connected at *fold points* via connector panels. An example appears in Fig. 8(b).

Preview. To compute the maximum reach, our algorithm computes the fold points and flat pieces. *Folding* the chain to *one* of these maximum configurations can then be done by sequentially rotating along the two hinges of the connector panel at each fold point, for angles that can be computed in constant time using basic spherical geometry. We remark that the algorithms are valid for a larger class of chains, described in the next section by a quite technical condition related to the triangle inequality on the sphere; it is much easier to follow and illustrate them for orthogonal chains.

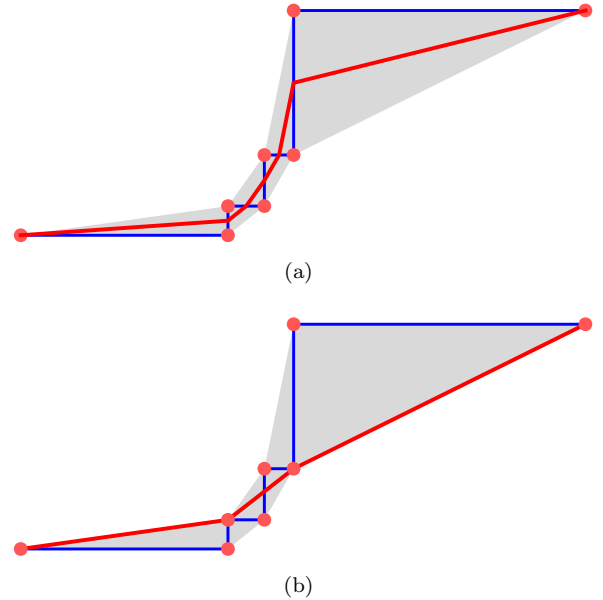


Figure 4: Finding the maximum reach for orthogonal chains. (a) The loose rope as a pseudoline inside the polygon. (b) The taut rope is the shortest path between the endpoints, and its length is the maximum reach.

Let us define the *polygon associated to a flat orthogonal chain in standard configuration* as the union of all triangles $p_i p_{i+1} p_{i+2}$, as in Fig. 1. Notice that the polygon interior, in gray in Fig. 1, is already triangulated by the chain *hinge segments* $p_i p_{i+1}$, $1 \leq i \leq n$.

Intuitively, imagine that we join the two endpoints by a loose rope and constrain it to meet all the hinge segments in *natural order*, as in Fig. 4(a), seeking to satisfy the condition of Theorem 3.1. In other words, we view the rope as a pseudo-line whose crossing pattern with the other lines is the natural order of the hinges. Then we pull the rope, while maintaining the chain flat and the rope confined inside the polygon. When the rope is taut, it becomes the *shortest (geodesic) path between the endpoints, inside the polygon*. If the path is a straight line, as in Fig. 3(a), it is the endpoint

axis intersecting the hinges in natural order. If it is not straight, it bends at some chain vertices, as in Fig. 4(b). These are the *fold points*. Formally:

ALGORITHM 1. Maximum reach and Fold Points

Input: A 3D orthogonal chain.

Output: The value of the maximum reach between the chain endpoints and the collection of fold points.

Method:

[1]. In linear time, lay the chain flat in the standard configuration and compute the associated polygon.

[2]. Compute the shortest (geodesic) path between the chain endpoints lying inside the polygon.

[3]. Output the length of the shortest path: this is the maximum reach.

[4]. Output the sequence of vertices on the shortest path: these are the fold points.

To compute the geodesic path, we can use, for instance, the linear algorithms of [11] or [7]. This is convenient since the polygon comes with the triangulation given by the edges of the orthogonal chain.

If we can show that there is a 3D realization of the chain in which the shortest path computed by this algorithm, marked in *red* on the panel-and-hinge chain, aligns to a straight-line red segment, then the correctness of this algorithm follows from Theorem 3.1. Those cases in which this property (of having a 3D realization as described above) also illustrate a more restricted version of Theorem 3.2, one where the endpoint axis meets the *hinge segments* in the natural order. This is not true for more general chains, as illustrated in Fig. 5. Indeed, our algorithm, specialized to achieve this stronger condition, will *not* detect the maximum in *all* panel-and-hinge chains, e.g. for the example in Fig. 5. In the next section we characterize the class of chain for which this stronger property holds, and show that it includes the orthogonal chains.

5 Computing a maximum-reach configuration

We move now to the problem of computing a configuration attaining the maximum reach. Expanding upon the intuitive description given in the previous section, at the position where the rope is taut, we *freeze* the lengths of its segments. Each frozen rope segment may cross some chain edges, which will stay flat in any maximal configuration. We use this observation to construct an associated panel-and-hinge chain as follows: the hinges (of the original standard flat chain) crossed by each frozen rope segment are themselves frozen flat, and their plane becomes a single new panel. The hinges of the new chain are the hinges incident to the fold vertices (two at each vertex). Note that our new panel-and-hinge chain also

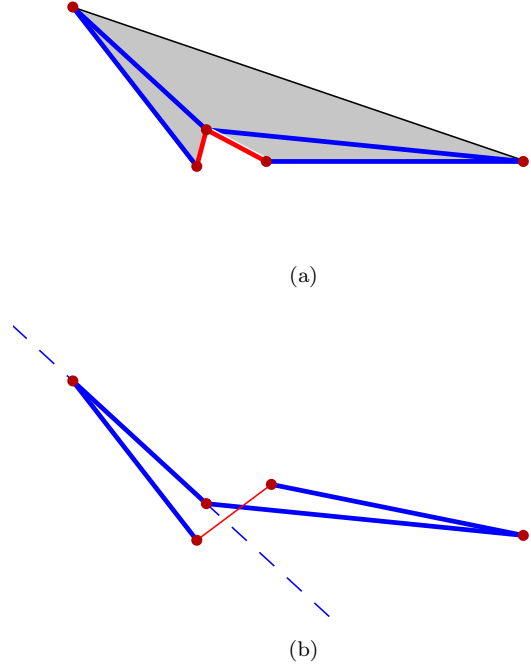


Figure 5: A (non-orthogonal) chain on which Algorithm 1 does not apply. (a) The chain, in a standard flat position, with the shortest path computed by the algorithm; this is not, however, the global maximum reach. (b) The maximum reach is attained in a flat, non-standard (with locally overlapping panels) position, in which the endpoint segment (red) meets the hinges in natural order, but one of them is crossed *outside* the *hinge segment* region (dashed).

contains the (planes of the) triangles between the two hinges at a fold vertex. See Fig. 6(b).

We now seek to fold this new chain from its flat position to a spatial configuration where the frozen rope segments are aligned. This may not be always possible (for arbitrary chains). But we show that we can always decide easily *when* it is so, by verifying a simple technical property (defined below). Finally, we prove that orthogonal chains *always* satisfy this property.

The antipodal triangle inequality on the sphere.

If we focus on one vertex of a panel-and-hinge chain, it has three panels and two hinges incident to it. This is visualized in Fig. 6(a). After the computation of the shortest path, a fold vertex is incident with two hinges and two segments of the frozen rope, as in Fig. 6(b).

At a fold vertex, the rope bends. This means that the three angles add up to more than π . The goal is to align the two rope segments by folding

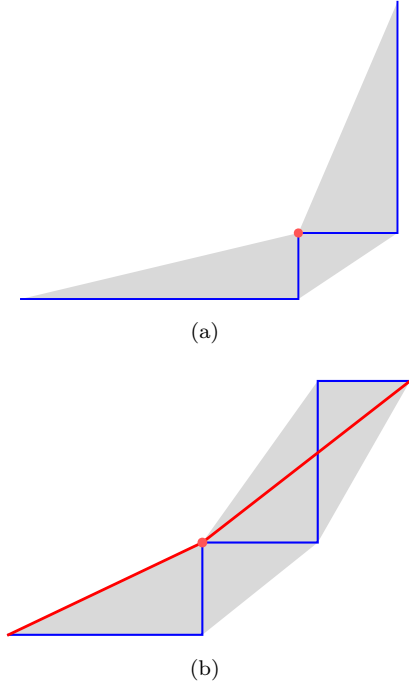


Figure 6: The three angles incident to a vertex, involved in the antipodal triangle inequality on the sphere, illustrated: (a) at a vertex of an orthogonal chain; (b) at a fold vertex, the three angles induced by the two red (frozen rope) and two blue (hinges) incident to the vertex. In this case, the two triangles crossed by the red line segment on the right become a new panel.

the simple three-edged *single vertex origami* of total spherical length between π and 2π (see [18, 14] for the relationship between spherical polygonal paths and single vertex origami). This cannot always be done, for instance when the three angles are 170, 30, 170 degrees. A necessary and sufficient condition is given by the following Lemma.

LEMMA 5.1. (Antipodal Triangle Inequality) *A spherical polygonal path of 4 vertices, made of three arcs of lengths a, b, c along the unit sphere, has a realization with antipodal endpoints iff the triplets of arc lengths $a, b, \pi - c$ and $\pi - a, b, c$ (and consequently also $a, \pi - b, c$) satisfy the triangle inequality.*

The spherical path with three arcs, in a position where its endpoints are antipodal, will be called an *antipodal spherical triangle*. See Fig. 7(b) for an example. The proof of the Lemma is elementary, since the triangle inequality must be satisfied for spherical triangles with edge lengths smaller than π , as well, and the antipodal triangle exists iff the complement $\pi - c$ of arc c (with respect to half of a great-circle)

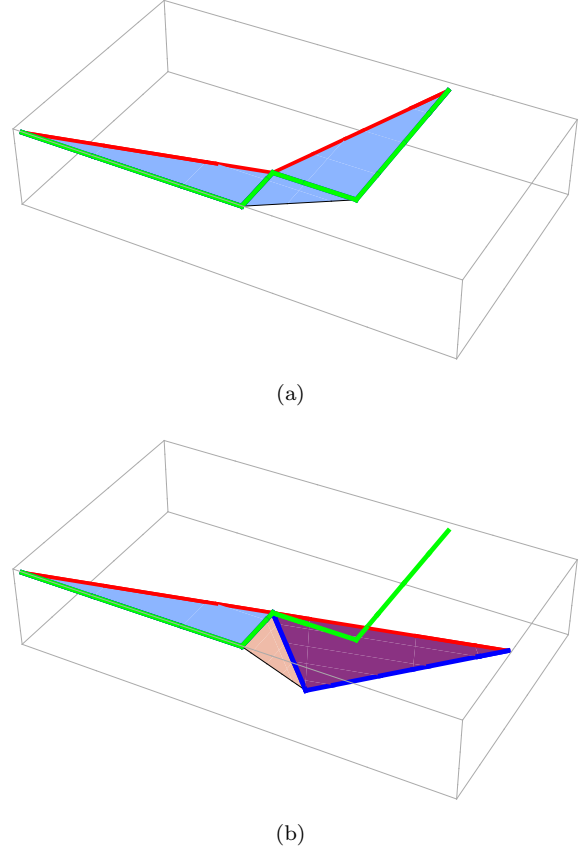


Figure 7: Folding three panels satisfying the Antipodal Triangle Inequality at a fold vertex. (a) The original flat configuration. (b) The final aligned fold. The taut rope in this position shows clearly that the maximum reach is obtained when its segments are aligned. The original flat chain contour is retained for visual reference.

forms a spherical triangle with the other two, iff the complement $\pi - a$ of arc a forms a spherical triangle with the other two. Elementary calculations show that the antipodal triangle inequality conditions lead to the following equivalent formulation:

COROLLARY 5.1. Antipodal Triangle Criterion *Three angles a, b and c , with $0 < a, b, c < \pi$ satisfy the antipodal triangle inequality iff they satisfy the following system of linear inequalities:*

$$(5.1) \quad a + b + c \geq \pi$$

$$(5.2) \quad a + b - c \leq \pi$$

$$(5.3) \quad a - b + c \leq \pi$$

$$(5.4) \quad -a + b + c \leq \pi$$

Using this criterion, we prove:

COROLLARY 5.2. *Chains with all equal angles α satisfy the Antipodal Triangle Inequality at every fold vertex iff $\alpha \geq \frac{\pi}{3}$. In particular, equal obtuse angles and orthogonal chains all fall into this category.*

Proof. We denote the three angles at a fold point as a, b and c , with b being the equal angle of the chain α , and use the criterion from Corollary 5.1. Notice first that $0 < a, c < \pi - b$: indeed, in the standard position of the chain, all sides follow just two directions (since the chain angles are equal), constraining the size of a and c to fall below $\pi - b$. The condition that the vertex is a fold point implies that the sum of the angles must exceed π , yielding condition (5.1). To verify (5.2) and (5.4), observe that $a + b - c < a + b < \pi - b + b = \pi$, and similarly for $-a + b + c$. Finally, $a - b + c < \pi - b - b + \pi - b = 2\pi - 3b$, which is $\leq \pi$ exactly when $b \geq \frac{\pi}{3}$.

THEOREM 5.1. (Maximum reach via rope inside polygon) *The maximum reach is the length of the end-to-end geodesic path inside the associated polygon iff each fold vertex satisfies the antipodal triangle inequality on the sphere. A maximum configuration is obtained by aligning the frozen rope segments at the fold points via folding the incident angle triplets to an antipodal spherical triangle.*

Proof. As we have observed in the Introduction, in any critical configuration the endpoint axis leads to the partitioning of the chain into flat pieces, connected at fold points (with a triangle in between). In a maximum reach configuration, the endpoint axis must meet the hinges in the natural order. At fold points, because of the alignment of the segments incident to the fold point, Lemma 5.1 applies. These statements hold in both directions for orthogonal chains.

COROLLARY 5.3. *The Maximum Reach of chains with all equal angles $\alpha > \frac{\pi}{3}$ is correctly computed by Algorithm 1.*

Finally, once we have computed the fold angles, we can fold the chain to a maximum configuration using a standard forward-kinematics robotics technique. For completeness, this is described in the next section.

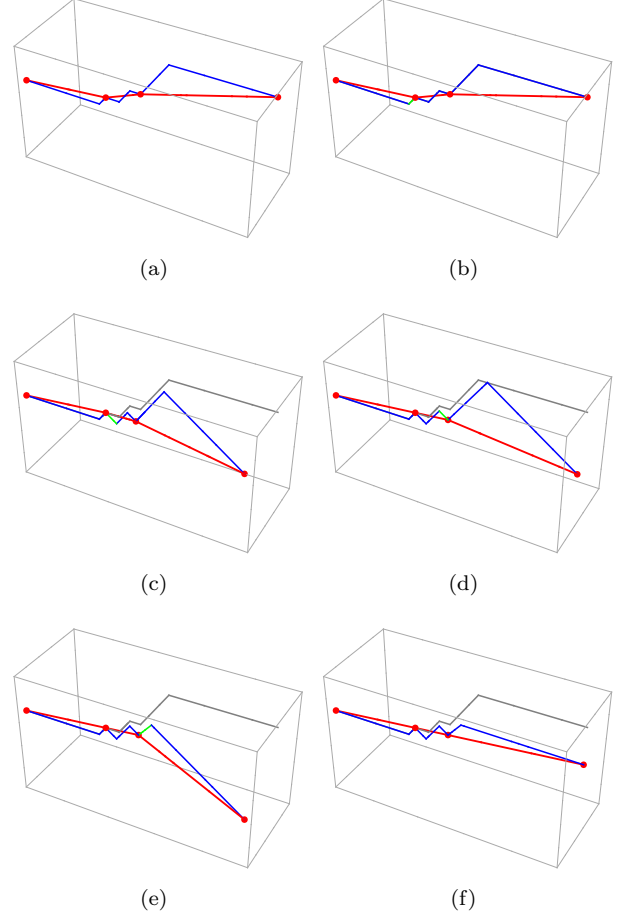


Figure 8: Folding a standard flat orthogonal polygon with two fold points to a maximum configuration. (a) The original flat configuration. (b)(c)(d)(e) With highlighted hinge, just before performing the corresponding rotation. (e) The final maximum fold. The original flat chain, in gray, is kept for visual reference. Notice the alignment of the rope segments in (d) and (f).

6 Motion Planning: folding to a maximum

The folding of the *single vertex origami* of three panels at each fold vertex into a position where the *frozen rope segments* become aligned is done sequentially via two rotations about the two hinges incident to the fold vertex. The alignment can be accomplished in one of two symmetric positions of the incident panels, which leads to an exponential number of possible configurations. Once we decide upon a desired fold pattern, then the entire folding process takes linearly many steps. Each folding step is a simple rotation of a part of the chain about one axis. The process of finding the final configuration becomes an instance of the classical forward kinematics problem for robotic manipulators.

ALGORITHM 2. Folding to Maximum Reach**Input:** A 3D orthogonal chain.**Output:** A 3D configuration of the chain in maximum reach position.**Method:**

[1]. Using Algorithm 1, compute the fold points and the position of the frozen rope.

[2]. For each fold point, compute the two dihedral angles at the incident chain hinges, corresponding to the alignment of the incident frozen rope segments. Decide which of the two local folds is to be chosen, and encode them as signs for the dihedral angles.

[3]. For each hinge incident to a fold point, rotate the part of the chain containing the terminus by the angle computed at step [2].

A few steps in the folding algorithm are illustrated in Fig. 8.

Analysis. Step [1] takes linear time, and step [2] takes time linear in the number of fold points. Thus the complexity of the folding process, computed in terms of number of folding steps, is linear.

THEOREM 6.1. *A maximal configuration can be reached in a linear number of vertex folding steps.*

Continuous folding. Step [3] of the algorithm can be adapted in several ways to create a continuous motion. First, the folding can be simulated continuously one hinge rotation at a time. Second, *all* the hinge rotations can be distributed at each time step, and applied at the same speed to obtain a folding trajectory.

None of these two motions guarantees that the endpoint distance increases monotonically. We describe next a different motion planning strategy, which achieves this property.

7 Expanding the endpoint distance to maximum

In the motion planning algorithm described above, the distance between the endpoints may not vary monotonically towards the maximum value. This can be observed for instance in the motion from in Fig. 8. It is natural to ask whether one can design such a specialized motion not by pursuing a gradient-based numerical method, but based on a discrete algorithm.

We next show a modification of the continuous version of the Algorithm 2 discussed in the previous section, which accomplishes this via a reduction to the planar Carpenter’s Rule problem.

THEOREM 7.1. *Starting from a planar standard configuration, a maximum-reach configuration can be attained in a manner that increases the endpoint distance throughout the motion.*

This is achieved by first applying the pseudo-triangulation roadmap algorithm of [17] to the *planar chain* determined by the frozen rope, in the plane of the first panel. The relative motion of two incident rope-segments is then used to determine the folding motion at each fold vertex.

ALGORITHM 3. Monotone Folding to Max Reach**Input:** A 3D orthogonal chain in a standard flat configuration, together with its fold points, frozen rope-segments and desired folding pattern at each fold vertex.**Output:** A trajectory that folds the chain to a maximum configuration and expands the endpoint distance throughout the motion.**Method:**

[1]. In the plane of the initial flat configuration of the chain, compute an expansive motion of the polygonal chain given by the frozen rope using the second author’s combinatorial algorithm [17] for the Carpenter’s Rule Problem based on pseudo-triangulations. The trajectory consists in continuous motion intervals (called expansive intervals) between two events which align two pseudo-triangulation edges.

[2]. For each expansive interval, compute the single-vertex origami motion at all fold-vertices.

The algorithm has a subtlety in Step [2], since even with one endpoint fixed and another moving along a determined spherical trajectory, the single vertex origami has an additional degree of freedom. At the end of the folding, that degree of freedom disappears, potentially causing some numerical instability. The pseudo-triangulation algorithm of [17] achieves the straightening of the frozen rope (and thus a maximal-reach configuration) in at most $O(n^3)$ reconfiguration steps.

8 Concluding remarks

Our very simple and efficient algorithms are the first ones that have a chance to make some aspects of the computations involved in folding of *special families* of chains (not too dissimilar from the actual protein backbones) tractable. As such, we anticipate that our techniques may open a new direction in the study of robot arms and their biomechanical applications. Although, for keeping the presentation uncluttered, we have formulated our algorithms for *generic* chains, it is not hard to extend them for other situations, such as

panel-and-hinge with more than four panels incident at one fold point; this extension is straightforward.

We conclude by formulating the following:

Conjecture: *There is a polynomial time, combinatorial algorithm for the Maximum Reach Problems, for general panel-and-hinge chains.*

We also conjecture that the problem is NP-hard for body-and-hinge chains, and emphasize that an NP-completeness would be an important theoretical advance for this case. So far no known methods, even approximate numerical ones, are guaranteed to compute the (generically unique) global maximum in this case: the gradient-based methods may get stuck in local maxima, and annealing methods may hop between local maxima with no criterion to guide them toward the global maximum. Note, however, that our natural-order criterion would allow these methods to decide, when in a local maximum, whether it is or not the global one.

Acknowledgment This research was sponsored by a DARPA “23 Mathematical Challenges” grant. All statements, findings or conclusions contained in this publication are those of the authors and do not necessarily reflect the position or policy of the Government. No official endorsement should be inferred.

References

- [1] Karim Abdel-Malek, Jingzhou Yang, and Yunqing Zhang. On the workspace boundary determination of serial manipulators with non-unilateral constraints. *Robotics and Computer-Integrated Manufacturing*, 24:60–76, 2008.
- [2] Karim Abdel-Malek, Harn-Jou Yeh, and Saib Othman. Interior and exterior boundaries to the workspace of mechanical manipulators. *Robotics and Computer Integrated Manufacturing*, 16:365–376, 2000.
- [3] Jorge Angeles. *Fundamentals of Robotic Mechanical Systems: Theory, Methods and Algorithms*. Mechanical Engineering Series. Springer-Verlag New York, Inc., 2007.
- [4] Ciprian S. Borcea and Ileana Streinu. Extremal configurations of manipulators with revolute joints. In *Reconfigurable Mechanisms and Robots, Proc. ASME/IFTOMM International Conference (REMAR’09)*, Jian S. Dai, Matteo Zoppi and Xianwen Kong (eds.), King’s College, London, UK, pages 279–284. KC Edizioni, June 2009. arXiv:0812.1375.
- [5] John F. Canny and David Parsons. Geometric problems in molecular biology and robotics. In *Proceedings of the Second International Conference on Intelligent Systems for Molecular Biology*, Stanford, CA. August 1994.
- [6] Marco Ceccarelli. A formulation for the workspace boundary of general n-revolute manipulators. *Mechanism and Machine Theory*, 31(5):637–646, 1996.
- [7] Bernard Chazelle. A theorem on polygon cuttings with applications. In *Proc. 23rd Ann. Symp. on Foundations of Computer Science (FOCS)*, pages 339–349, 1982.
- [8] Stephen Derby. The maximum reach of revolute jointed manipulators. *Mechanism and Machine Theory*, 16(3):255–261, 1981.
- [9] Joseph Duffy. *Analysis of Mechanisms and Robot Manipulators*. Wiley, New York, 1980.
- [10] Joseph Duffy and M. J. Gilmartin. Limit positions of four-link spatial mechanisms: mechanisms having revolute and cylindric pairs. *Journal of Mechanisms*, 4:261, 1969.
- [11] Leonidas J. Guibas, John Hershberger, Daniel Leven, Micha Sharir, and Robert Endre Tarjan. Linear-time algorithms for visibility and shortest path problems inside triangulated simple polygons. *Algorithmica*, 2(4):209–233, 1987.
- [12] James Urey Korein. *A geometric investigation of reach*. MIT Press, Cambridge, MA, June 1985.
- [13] A. Kumar and K. J. Waldron. The workspaces of a mechanical manipulator. *Journal of Mechanical Design*, 103(3):665–672, 1981.
- [14] Gaiane Panina and Ileana Streinu. Flattening single-vertex origami: the non-expansive case. In *Proc. 25th Annual ACM Symposium on Computational Geometry*, pages 316–323, June 2009.
- [15] Robert J. Schilling. *Fundamentals of Robotics*. Prentice Hall, 1990.
- [16] R. G. Selfridge. The reachable workarea of a manipulator. *Mechanism and Machine Theory*, 18(2):131–137, 1983.
- [17] Ileana Streinu. Pseudo-triangulations, rigidity and motion planning. *Discrete and Computational Geometry*, 34(4):587–635, November 2005.
- [18] Ileana Streinu and Walter Whiteley. Single-vertex origami and spherical expansive motions. In Jin Akiyama and M. Kano, editors, *Proc. Japan Conf. Discrete and Computational Geometry (JCDCG 2004)*, volume 3742 of *Lecture Notes in Computer Science*, pages 161–173, Tokai University, Tokyo, 2005. Springer Verlag.
- [19] K. Sugimoto and Joseph Duffy. Determination of extreme distances of a robot hand - Part 1 : A general theory. *Journal of Mechanical Design*, 103(3):631–636, 1981.

Flattening Single-vertex Origami: the Non-expansive Case

Gaiane Panina¹ and Ileana Streinu²

Abstract

A *single-vertex origami* is a piece of paper with straight-line rays called creases emanating from a fold vertex placed in its interior or on its boundary. The *Single-Vertex Origami Flattening* problem asks whether it is always possible to reconfigure the creased paper from any configuration compatible with the metric, to a flat, non-overlapping position, in such a way that the paper is not torn, stretched and, for *rigid origami*, not *bent* anywhere except along the given creases.

Streinu and Whiteley showed how to reduce the problem to the carpenter's rule problem for spherical polygons. Using spherical expansive motions, they solved the cases of open $< \pi$ and closed $\leq 2\pi$ spherical polygons. Here, we solve the case of open polygons with total length between $[\pi, 2\pi)$, which requires non-expansive motions. Our motion planning algorithm works in a finite number of discrete steps, for which we give precise bounds depending on both the number of links and the angle deficit.

1 Introduction

In this paper, we answer in the affirmative the following conjecture of [14]:

A single-vertex origami whose fold vertex is placed on the boundary of the paper can always be reconfigured to the flat position with a non-colliding continuous motion.

The technical formulation of the problem is given below.

Rigid Origami. An origami is a flat piece of paper marked with a straight-line plane graph drawing. Fig. 1 exemplifies the first non-trivial type, which is the topic of this paper: an origami with just a single vertex. By creasing the paper along the edges and possibly bending the paper while maintaining its intrinsic metric (i.e. not allowing any tearing or stretching), the origami will take various 3D shapes. *Rigid origami* is the study of those configurations and motions which further restrict the faces to remain planar. Thus they behave like rigid panels hinged along the crease lines, along which they may rotate. While in practice the paper bends during folding, this model offers a rigorous mathematical formulation and potential for algorithmic treatment.

In the quest of mathematical laws for origami folding, we start with the simplest situation: the **single-vertex origami**. This is the case of a single vertex, with edges emanating from it that partition the paper into wedges, as in Fig. 1(a).

Non-Flat Paper Folding. It is worth mentioning that the abstract single-vertex origami problem includes *non-flat paper*, looking very much like the corner of a polyhedral surface, rather than a flat sheet. The angle sum around the vertex may be smaller than 2π , as in the case of a convex

¹Institute for Informatics and Automation, V.O. 14 line 39, St.Petersburg 199178, Russia. panina@iiias.spb.su

²Department of Computer Science, Smith College, Northampton, MA 01063, USA. istreinu@smith.edu.

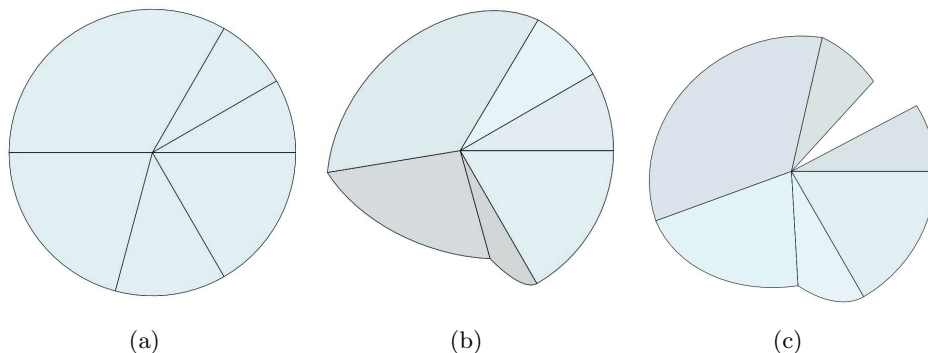


Figure 1: (a) Single vertex origami with interior fold vertex, in a flat non-overlapping configuration and (b) in a 3D folded configuration. For contrast, if a crease in the origami from (a) is cut up, then the fold vertex is no longer interior; a 3D folded shape of this case is illustrated in (c).

polyhedron vertex, or larger than 2π , as in saddle surfaces and hyperbolic virtual polytopes [8]. It equals 2π for an interior-vertex origami as in Figures 1(a), 1(b).

Folding rigid origami. Very little is known mathematically about rigid origami. According to T. Hull’s web page [4] devoted to the topic, as of 2003 only two papers have been published in this area. The classical origami literature is concerned mostly with characterizations of folded states and axiomatics for folding patterns. Equally interesting and important, but also very little studied is the motion planning problem for origami, i.e. the design of reconfiguration trajectories. In particular, the non-self-intersecting foldability and reconfiguration of rigid origami has received so far *very* little attention, mostly because it is a very difficult problem. Section 12.3 of [3] summarizes in a little over one page what is known about continuous foldability of single-vertex origami, which is essentially the previous paper [14] of the second author and Whiteley.

Meanwhile, the topic gained momentum due to new robotics applications [7, 1] and the advent of practical nano-origami at the DNA [10] and mechanical [6, 5, 11] level. In simulations, the probabilistic roadmap algorithm has been shown to find folding trajectories for small “paper craft” puzzles [12]. Understanding the laws of origami *folding* is becoming a recognized area of mathematical and algorithmic research, as the field itself moves from recreational aspects to increasingly practical applications.

Single-vertex Origami. One of the main questions in algorithmic origami is: *Are there origami folded shapes which are compatible with the creases and the induced metric of the paper, but which cannot be folded by a collision-avoiding motion?* Here we show that there aren’t any single-vertex ones. Moreover, the reconfiguration of a single-vertex origami between two configurations can be performed algorithmically, in finite time.

For a vertex placed in the interior of the piece of paper, as in Fig. 1(a), or for a boundary vertex incident to a paper angle¹ of at most π , as in Figures 2(a) and 2(c), Streinu and Whiteley [14] answered the problem in the affirmative by reducing it to a version of the carpenter’s rule problem [2, 13] extended from planar to spherical polygons. The method used in [14], based on expansive motions (in particular those induced by pointed spherical pseudo-triangulation mechanisms), can be applied to unfold closed spherical polygons of total length less than 2π , and open ones less than π (precise definitions of single-vertex origami angle size and spherical polygon length will be

¹Only the angles incident to the fold vertex are relevant to the question.

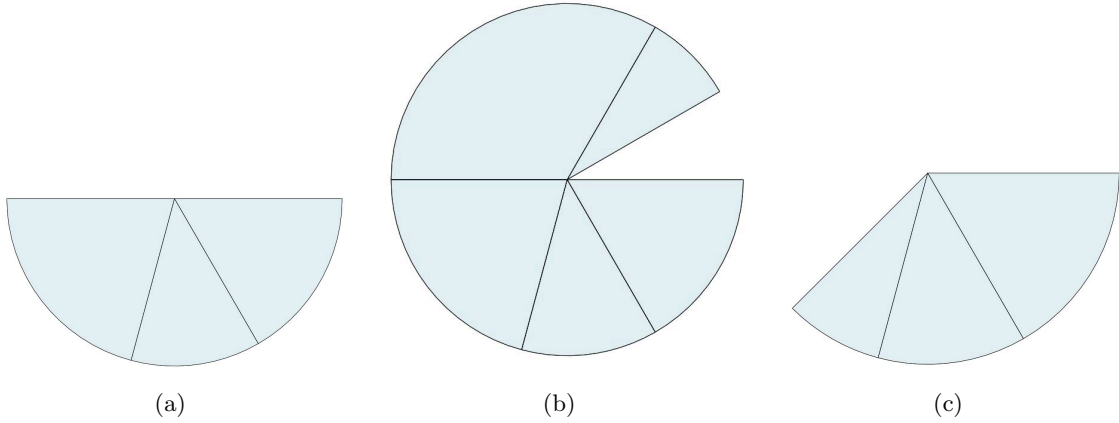


Figure 2: Single-vertex origami: the *vertex-on-boundary* case. The fold vertex is placed: (a) on an edge of the paper, (b) at a reflex corner and (c) at a convex corner.

given in Section 2). The algorithm uses at most $O(n^3)$ steps, each one being induced by the well-defined expansive direction of motion of a pseudo-triangulation one-degree-of-freedom mechanism. The paper [14] also showed that spherical polygons of lengths larger than 2π (and, equivalently, single-vertex origamis of total angle larger than 2π) may not be reconfigurable between any two configurations.

The remaining case, of open polygons whose length lies between π and 2π , or single-vertex origami incident to a reflex paper corner as in Fig. 2(b), is not directly amenable to the expansive motion and the pseudo-triangulation techniques, as it requires both contractive and expansive motions.

In this paper we settle the problem for spherical bar-and-joint polygonal paths of total length $\alpha \in (\pi, 2\pi)$, by showing that it is always possible to unfold them without self-collisions. The motion (necessarily partially non-expansive) can be carried out in discrete steps, and completed in finite time, for which we give precise bounds. However, the bound on the number of steps will depend not just on n (the number of links in the chain), but also on the angle deficit $2\pi - \alpha$.

2 Preliminaries

We start by introducing the relevant concepts, and we summarize the previous results which reduce the single-vertex origami problem to the spherical carpenter's rule. Then we give a brief summary of the techniques we rely on for establishing the main result of the paper, as well as an overview of concepts from spherical geometry needed in our proofs.

2.1 From origami to spherical chains

Single-vertex origami. A *single vertex origami* is a bounded or unbounded piece of paper, together with a point on it (the *fold vertex*) and a finite set of rays emanating from the vertex, called *creases*. The rays induce a natural ordering around the vertex. If the vertex is placed in the interior of the piece of paper, as in Fig. 1(a), then this is the circular counter-clockwise (ccw) ordering of the rays around the vertex. If the vertex is placed on the boundary, as in Fig. 2(b), then

the ordering is linear and starts and ends at an edge along the paper’s boundary. These extreme edges are not creases.

Single-vertex panel-and-hinge chains. We consider only *rigid origami*, where the wedge-like flat regions between two consecutive creases maintain their intrinsic and extrinsic metric, i.e. they behave like rigid flat metal *panels* rather than flexible paper. Single-vertex origami can now be modeled as a collection of rigid polygons (the panels) connected by hinges (the creases), such that all the hinges are concurrent in a single vertex.

For single-vertex origami, the shape of the polygonal panels is irrelevant to questions of self-intersection. They may even be unbounded. All that matters is where the vertex is placed: in the interior of the paper, or on its boundary. In the first case, the panel-and-hinge structure forms a *closed* chain. In the second, it is an *open* chain.

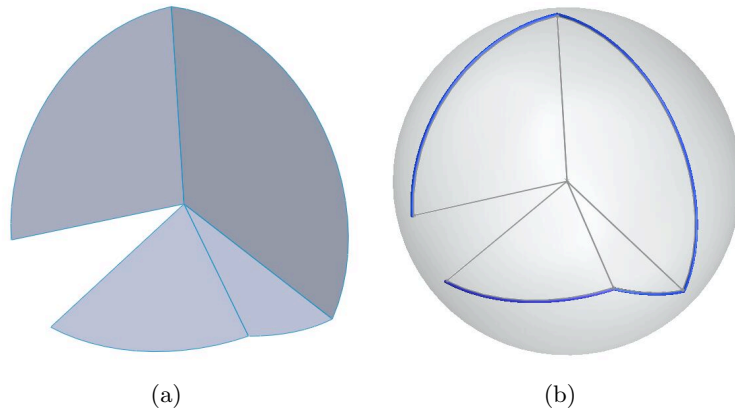


Figure 3: Two views of an open single vertex origami, in a 3D folded position: (a) as a panel-and-hinge chain, and (b) as a spherical polygonal chain.

Spherical chains and polygons. We cut a circle (of sufficiently small radius so that it crosses all the crease lines), centered at the vertex. From now on, we work with this bounded piece of paper, as in Figures 1 and 2. Each panel is now bounded by two straight line edges (corresponding to either the creases, or to the paper boundary) and a circular arc. In any configuration of the origami in three-dimensional space (3D), these circular arcs are arcs of great-circles on a sphere (assumed to be the unit sphere) centered at the fold vertex.

When the fold vertex is on the boundary of the piece of paper and the spatial origami configuration does not bring (or glue) the boundary edges together, the arcs will form a spherical polygonal path or chain. When the vertex is interior to the original piece of paper, or the folding glues boundary edges, then it will be a closed polygon, as in Fig. 1(b). Notice that the panels intersect if and only if their corresponding circular arcs intersect on the sphere.

From now on, we work with the spherical polygonal chain model, and assume we are on the unit sphere.

Notation. A spherical polygonal chain with n edges is given by an ordered set of points on the unit sphere $p = \{p_0, p_1, \dots, p_n\}$. Its edges are denoted by $e_i = (p_{i-1}, p_i)$. A *subchain* $p[i : j]$ consists in all vertices and edges between the points p_i and p_j . Closed polygons appear only indirectly in this paper, via references to previous work, so we do not introduce any special notation for them.

Arc length. A spherical arc has a length, which is measured by the angle at the center of the

sphere between the two rays that span the arc. A *short arc* has length less than π . A *long* one has length larger than π , and less than 2π . Arcs larger than 2π are self-overlapping and hence not within the scope of this paper. All throughout, we work only with short arcs.

Chain length. The *length* of a spherical polygonal chain is the sum of its arc lengths. We distinguish three categories of chains: *short*, of length strictly less than π , *medium* (the case considered in this paper), of length between $[\pi, 2\pi)$, and *large*, those exceeding 2π .

Configuration space. The set of all the possible positions of the chain vertices which are compatible with the given edge lengths, up to spherical rigid motions (rotations around the center) is called the *configuration space* of the chain. To eliminate the rigid motions, we can *pin down* any edge.

Flat, hemispherical and sphere-spanning chains. A chain configuration stretched along a great-circle will be called *flat*. If it is contained in some open hemisphere, we call it *hemispherical*. Otherwise it is called *sphere-spanning*. For example, any closed or open polygonal chain of length at most π is hemispherical. When the length exceeds π , some configurations may be hemispherical, others may not.

2.2 Unfolding spherical chains: previous results

Hemispherical versus planar chains. For points and edges lying in a hemisphere, in particular for a hemispherical chain, oriented matroid concepts such as pointedness and convex hulls are in one-to-one correspondence with their planar counter-parts. This allows us, among others, to define spherical pointed pseudo-triangulations and apply all the results on polygon unfolding from [13]. We refer the reader to that paper, or to the survey [9] for background material on pointed pseudo-triangulations. These concepts are not needed here, because we will use Streinu’s pseudo-triangulation-based unfolding algorithm [13] only as a black box.

Unfolding single-vertex origami: summary of previous results. There are two types: (a) *long chains* (open or closed) may *not* be reconfigurable, and (b) *short chains* and *medium closed chains* can always be reconfigured. For closed chains, the reconfiguration is carried out in the same orientation class. The main idea is that short chains and medium polygons are confined to a hemisphere. In this case, a theorem of [14] establishes an equivalence between infinitesimal motions of hemispherical and planar polygons, which transfers all the results of the planar carpenter’s rule problem to the spherical setting. This equivalence is at the infinitesimal level, and holds whenever a chain configuration can be confined to some hemisphere. It does not require that the total length of the chain be short. We will make substantial use of this observation.

When is a chain hemispherical? Short chains and medium polygons are *always* confined to a hemisphere: this property seems obvious, but its formal proof has not appeared before. It is a simple consequence of our Separation Theorem (theorem 5 in Section 4). In the planar carpenter’s rule problem, an open chain was treated by just closing it with additional edges. On the sphere, we cannot add edges without increasing the total length.

In all the other cases, some chain configuration *may* span the sphere. If it does not, the motion can proceed, expansively, until the chain touches a great-circle in at least three points.

The crux of our argument is the treatment of the case of a medium-length chain in a configuration which spans the sphere. The following classical concepts will be needed.

2.3 Spherical polar-duality

Computational Geometers are familiar with planar dualities between points and lines, and their incidence and orientation preserving properties. Here, we make use of their spherical counterpart, which has even stronger, measure-theoretic properties on which we rely in our proofs.

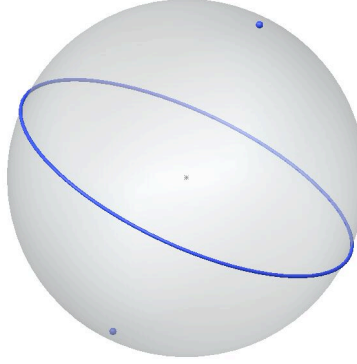


Figure 4: The point-to-great-circle duality on the sphere.

Spherical polar-duality. The well-known duality between great-circles and antipodal pairs of points takes a great-circle c (viewed as an *equator*) to a pair of antipodal points called its *poles*, as in Fig. 4. The poles are the intersection points of the sphere with the line orthogonal to the supporting plane of the great circle, and going through the center of the sphere.

This duality (usually referred to as a polar-duality) has all the good incidence and orientation-preserving properties of planar dualities familiar to Computational Geometers, and more. In fact, the *natural* definition of point-line duality is the spherical version. We will make use of the following properties.



Figure 5: A spherical digon, or lune. The two antipodal points are connected by two arcs of length π . The area of the lune equals twice the length of the arc it spans on the great circle orthogonal to both arcs.

The antipodal pair of points where two great-circles cross is dual to a great-circle. This passes through two pairs of antipodal points on the sphere, which are dual to the two original crossing great-circles. These two pairs of points determine four *short* arcs on their spanning great-circle,

grouped into two antipodal pairs. Two great-circles determine four *lunes*, grouped into two antipodal pairs. The duality takes a short arc (and its antipodal) to a lune (and its antipodal). The set of circles crossing an arc (and its antipodal) is mapped, by duality, to a set of points contained in the dual lune, and its antipodal. The duality allows for the definition of a measure on sets of great-circles.

Lebesgue measure on the sphere. The set of all great circles is endowed with Lebesgue measure μ . This is the spherical area of the dual set of antipodal points of the great circles, divided by 2 to account for the antipodal symmetry. The measure of all the great circles is thus 2π , half of the total area 4π of the unit sphere.

Lemma 1 *The Lebesgue measure of the set of great circles crossing an arc of length $\alpha < \pi$ equals 2α .*

Proof. The dual of the set of circles crossing an arc is a lune of span α and its antipodal. The area of a lune is proportional to the fraction of a great circle spanned by twice its spanned arc, hence 2α . See Fig. 5. $\square$

Belts. A *belt* is the area between two circles at equal distance ℓ from a great circle, called its *median* equator (or great circle). The width $w = 2\ell$ of the belt is the length of the arc orthogonal to its median equator. The following lemma allows us to measure the set of great-circles contained in a belt. The circles it refers to are arbitrary circles on the sphere, not great-circles. See Fig. 6.

Lemma 2 *The polar-dual of a belt of width w is a pair of antipodal circles of diameter w .*

Proof. Let us denote by c a great-circle and by p_c one of its poles. A great circle c contained in the belt region makes an angle of at most $\frac{w}{2}$ with the belt's equator e . Its dual p_c lies at a spherical distance at most this value from the dual p_e of the equator, i.e. it lies on a circle of diameter w . $\square$

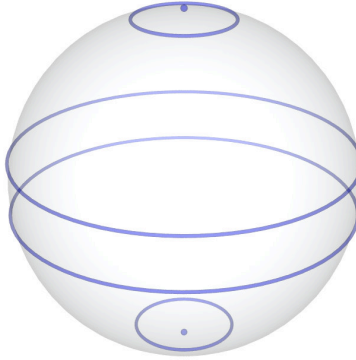


Figure 6: The dual of a belt around an equator is a pair of circles centered at the two poles of the equator.

Convex spherical polygons. A spherical (closed) polygon is *convex* if for each edge, its spanning great circle contains all the other polygon vertices and edges in one of its two hemispheres. In particular, a lune is a convex spherical polygon. A convex spherical polygon is contained in a hemisphere. Hence the convex polygon has a well-defined interior, which is the region lying in the hemisphere.

The following lemma gives a useful estimate of the area of a spherical convex polygon. It is worth noticing that this has no counter-part in the Euclidean setting.

Lemma 3 *Let $K \subset S^2$ be a (spherically) convex polygon and $\text{area}(K)$ its area. Let d be the maximal diameter of a circle lying in K . Then:*

$$\text{area}(K) \leq 2d$$

Proof. The statement is true with equality for a lune, whose area is twice the diameter of a maximum inscribed circle, of diameter equal to the arc spanned by the lune.

For an arbitrary polygon, the maximum inscribed circle can be either tangent to two sides, or to more. In the first case, if we eliminate all edges of the polygon not tangent to the circle, we increase the area, and get a lune, for which the statement is valid. In the other case, we eliminate all edges not tangent to the circle, and all but three of those which are tangent. We get a spherical triangle and a circle inscribed in it. Now rotate one of the sides, keeping it tangent to the circle, until it makes a lune with one of the remaining sides, for which the circle is the minimum inscribed one. In the process, the area has increased but the diameter of the circle remained the same. $\square$

3 Main result: flattening medium chains

We describe now an algorithm for planning the motion of a medium-size spherical chain from an arbitrary configuration to one lying on the equator. We know that a fully expansive global motion may not be possible in this case: indeed, if the endpoints of the chain lie more than $2\pi - \alpha$ apart, where α is the length of the chain, then they must get closer together to reach the final configuration, where they lie at exactly this distance. To design a non-self-intersecting unfolding trajectory, we will patch together segments of expansive trajectories (which are possible when the chain lies in a hemisphere) with other trajectories that are expansive only for subchains. These subchains will be identified by a *separating great circle* cutting through exactly one edge (we will show later that it always exists). Since each of the two parts now lie in separate hemispheres, they can be expanded there, independently. When the unfolding of one of these subchains reaches the boundary of the hemisphere containing it, we recalculate the separating great circle, and continue. Formally:

Algorithm: Straightening a Spherical Chain of Medium Length

Input: A spherical chain p of total length α between π and 2π .

Output: A trajectory to unfold the chain onto a great-circle.

Method:

1. If the chain is straightened onto a great-circle, then stop.
2. If the chain lies in a hemisphere, apply an expansive unfolding motion (e.g. Streinu and Whiteley's [14] adaptation to the hemisphere of the pseudo-triangulation algorithm of [13]). Continue for as long as the chain remains hemispherical.
3. Otherwise, find an edge e_k of the chain such that the measure of all separating great-circles cutting through the edge is the largest. The edge splits the chain into two parts $p' = p[0 : k - 1]$ and $p'' = p[k : n]$.

Fix a belt b of width $w \geq (2\pi - \alpha)/(n + 2)$ cutting through the edge e_k , as in Fig. 7. Choose its median as the boundary great-circle (equator) separating the chain into two parts, each lying in its own hemisphere. The edge e_k cuts through the equator defining these hemispheres.

In each hemisphere, consider only the part of the edge e_k lying in it as part of the hemispherical subchain. Pin this edge, and prepare to proceed with the pseudo-triangulation expansive algorithm applied only for one of the subchains, in its hemisphere.

4. Perform the hemispherical expanding unfolding process (as in step 2) for one of them. Stop when either that subchain is straightened or when it hits the middle equator. Then repeat from Step 1.

□

We emphasize that Step 2 is carried out only *for the part of a chain that lies in a hemisphere*. We remind the reader that the pseudo-triangulation algorithm needs a pinned-down edge. We will use the separating edge for this purpose, or, to be precise, the part of it that lies in the hemisphere where the expansive motion takes place.

The existence of the separating belt of the specified width is proven in Section 4.

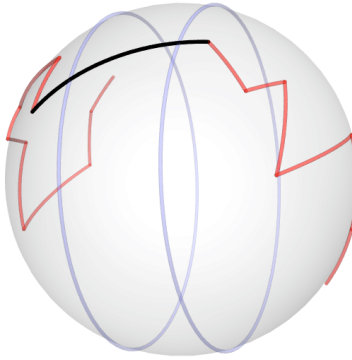


Figure 7: A sphere-spanning chain and a separating belt. The separating edge is shown thicker than the other polygon edges.

Time analysis. The algorithm works in phases, which are expansive on all or part of the chain. A phase is the trajectory between two switches of the separating edge and its belt. We know from [13] that each expansive motion lasts at most $O(n^3)$ reconfiguration steps. To complete the time analysis we need to bound the number of phases.

Proposition 4 (Finiteness of the algorithm) *A chain p of length α with n edges will be straightened in at most $\frac{2\pi\alpha}{(2\pi-\alpha)}(n+2)(n-1)$ phases.*

Proof. Let β_i be the small angle between the edge e_i and e_{i+1} , for $i = 1, \dots, n-1$. We will use $\Delta = \Delta(p)$, the sum of these small angles at the inner vertices, as a measure of progress of the algorithm:

$$\Delta = \sum_{i=1}^{n-1} \beta_i$$

First, notice that Δ only grows during the unfolding process. For a straightened chain, it achieves its maximum at $\Delta = (n-1)\pi$. Therefore it suffices to show that during each phase, Δ increases with a positive fraction.

Consider a phase that moves the chain from configuration p to q . Define $\delta = \Delta(q) - \Delta(p)$.

The definition of the phase and the choice for the width of the belt implies that there exists a vertex j such that its distance d between its original position in p and its final position in q is at least $w/2$, where w is the width of the belt. Indeed, originally the belt was vertex-free, and at the end some vertex hits its middle line.

It is easy to see that $d \leq \alpha\delta$. Indeed, $\alpha\delta$ is the distance traveled, on a great circle lying on the sphere, by a point rotating by an angle of α about a center of rotation at distance δ from it. This certainly exceeds the distance d by which point j was displaced. Combining these inequalities we obtain that $w/(2\alpha) \leq \delta$. Since the belt width was taken to be at least $w \geq \frac{2\pi-\alpha}{n+2}$, the number of phases is at most:

$$\frac{(n-1)\pi}{\delta} \leq \frac{2\pi\alpha(n-1)}{w} \leq \frac{2\pi\alpha}{(2\pi-\alpha)}(n+2)(n-1)$$

□

The rest of the paper contains the proofs.

4 Proofs

4.1 Separating the chain

The main technical tool is the existence of a separating great-circle, as needed by Step 2 of the algorithm. More precisely, we need a set of separating circles of large Lebesgue measure.

Theorem 5 (Separation by great-circle) *Let $p = (p_0, \dots, p_n)$ be an open spherical chain of total length between π and 2π . Then, there exists a great circle cutting through at most one edge of the spherical chain.*

Proof. Let $p = \{p_0, \dots, p_n\}$ be an open spherical chain with n arcs $e_i = (p_{i-1}, p_i)$ of lengths α_i , and total length $\alpha := \sum_{i=1}^n \alpha_i$.

Consider the set $\mathcal{C}$ of all great circles. Its Lebesgue measure is 2π , the area of a hemisphere.

Let $\mathcal{C}_i$ be the set of all great circles that cross the edge e_i , for $i = 1, \dots, n$. Consider the subset $\mathcal{N}$ of *nice* great-circles that intersect the chain in no more than one point. We partition them into equivalence classes $\mathcal{N} = \mathcal{N}_0 \cup \mathcal{N}_1 \cup \dots \cup \mathcal{N}_n$, where $\mathcal{N}_0$ is the set of circles that do not cross any edge, and $\mathcal{N}_i$ is the set of those crossing only edge e_i . We have $\mathcal{N}_i = \mathcal{C}_i \cap \mathcal{N}$, $i = 1, \dots, n$ and $\mathcal{N}_0 = (\mathcal{C} \setminus \cup_{i=1}^n \mathcal{C}_i) \cap \mathcal{N}$.

Some of these $\mathcal{N}_i$ classes may be empty. We want to show that at least one is not empty, and that it has a sizeable measure. Since we use the Lebesgue measure of these sets, which is a measure of area, we can ignore the circles passing through the vertices, which account for lower dimensional subsets.

The theorem now follows from Corollary 7 below to the following Lemma, which gives a lower bound for the Lebesgue measure $\mu(\mathcal{N})$. □

Lemma 6 (Lebesgue measure of nice great circles) *For any set $e = \{e_1, \dots, e_n\}$ of n arcs on the sphere, of total length α , the Lebesgue measure $\mu(\mathcal{N}) = \sum_{i=0}^n \mu(\mathcal{N}_i)$ of the set of nice great circles $\mathcal{N}$ satisfies the inequality:*

$$\sum_{i=1}^n \mu(\mathcal{N}_i) + 2\mu(\mathcal{N}_0) \geq 2(2\pi - \alpha)$$

Before giving the proof, we observe two straightforward corollaries.

Corollary 7 (Nice great circles exist) *For any set of n arcs on the sphere, of total length $\alpha \leq 2\pi$, either there exists one great circle which doesn't cross any of the arcs, or it crosses exactly one of them.*

Corollary 8 (There exist many nice great circles) *For any set of n arcs on the sphere, of total length $\alpha < 2\pi$, one of the nice sets of great circles has large Lebesgue measure $\mu(\mathcal{N}_i) \geq (4\pi - 2\alpha)/(n + 2)$.*

Proof of Lemma 6. We integrate, over the set of all great circles with Lebesgue measure μ , the function $\#(c \cap e)$ giving the number of crossings of a great circle c with the set of arcs e . We obtain:

$$\int_{\mathcal{C}} \#(c \cap e) d\mu(c) = \int_{\mathcal{N}} \#(c \cap e) d\mu(c) + \int_{\mathcal{C} \setminus \mathcal{N}} \#(c \cap e) d\mu(c)$$

First, notice that over the set $\mathcal{N}$, there is at most one crossing.

$$c \in \mathcal{N}_0 \implies \#(c \cap e) = 0 \implies \int_{\mathcal{N}_0} \#(c \cap e) d\mu(c) = 0$$

$$c \in \mathcal{N} \setminus \mathcal{N}_0 \implies \#(c \cap e) = 1 \implies \int_{\mathcal{N} \setminus \mathcal{N}_0} \#(c \cap e) d\mu(c) = \mu(\mathcal{N} \setminus \mathcal{N}_0)$$

Over the rest $\mathcal{C} \setminus \mathcal{N}$, there are at least two crossings:

$$c \in \mathcal{C} \setminus \mathcal{N} \implies \#(c \cap e) \geq 2 \implies \int_{\mathcal{C} \setminus \mathcal{N}} \#(c \cap e) d\mu(c) \geq 2\mu(\mathcal{C} \setminus \mathcal{N}) = 2(2\pi - \mu(\mathcal{N}))$$

The integral over all great circles is the total length of the set of arcs:

$$\int_{\mathcal{C}} \#(c \cap e) d\mu(c) = 2\alpha$$

Putting everything together:

$$2\alpha \geq \mu(\mathcal{N} \setminus \mathcal{N}_0) + 2(2\pi - \mu(\mathcal{N}))$$

Finally, using $\mu(\mathcal{N} \setminus \mathcal{N}_0) = \sum_{i=1}^n \mu(\mathcal{N}_i)$ we get:

$$\sum_{i=1}^n \mu(\mathcal{N}_i) + 2\mu(\mathcal{N}_0) \geq 2(2\pi - \alpha)$$

Since the sum on the lefthand side has at most $n + 2$ non-zero parts, there must exist an equivalence class $\mathcal{N}_i$ whose Lebesgue measure is large: $\mu(\mathcal{N}_i) \geq (4\pi - 2\alpha)/(n + 2)$. $\square$

We have thus shown that for any set of edges, in any placement on the sphere, of total sum strictly less than 2π , there exists a large class of *nice* great circles.

The lower bound on the Lebesgue measure of great circles is used next to compute the width of a belt, which in turn was used to bound the number of reconfiguration steps (step 4) in the Algorithm.

4.2 Wide belts

Finally, we give the bound on the width of a belt chosen to mark the progress made during a phase of the algorithm.

We start with a straightforward observation. Recall that β_i is the small angle between the edge e_i and e_{i+1} , for $i = 1, \dots, n-1$.

Proposition 9 *During this unfolding process, the angles β_i never decrease.*

Proof. Indeed, the motion is expansive within one hemisphere, so all the angles inside a hemisphere increase. And all the vertices lie inside one hemisphere or the other. $\square$

We are ready to bound the belt width. Let p be an arbitrary chain configuration. There are two cases to consider: (a) if the chain is not contained in a hemisphere, or (b) if it is. In the first case (a), from Lemma 6 we obtain: $\mu(\mathcal{N}_0) = 0$ and at least one of the classes $\mathcal{N}_i$ is large, $\mu(\mathcal{N}_i) \geq \frac{4\pi-2\alpha}{n}$. But we do not need to be so precise. In either case, we can safely bound $\mu(\mathcal{N}_i) \geq \frac{4\pi-2\alpha}{n+2}$.

Lemma 10 (Belt bounds)

1. *If $\mathcal{N}_0$ is a large equivalence class, then there exists a belt of width at least $w \geq (2\pi - \alpha)/(n+2)$ which does not intersect the chain r .*
2. *If $\mathcal{N}_i, i \neq 0$ is a large equivalence class, then there exists a belt of width $w \geq (2\pi - \alpha)/(n+2)$ which does not intersect the chain except for the edge e_i .*

Proof. The two cases are similar, so we do only one. Define the set of points dual to the great circles in $\mathcal{N}_i$ as $\mathcal{N}_i^* = \{c^* : c \in \mathcal{N}_i\}$. It is easy to see that it is a convex spherical polygon. Its area is at least $(4\pi - 2\alpha)/(n+2)$. Therefore, applying Lemma 3, it follows that it contains a circle C of diameter $w = (2\pi - \alpha)/(n+2)$. Then the great-circles dual to points in C sweep a belt $C^* \subset \mathcal{N}_i$ of width w . $\square$

This completes the proof.

5 Concluding remarks and open questions

We have shown that there exists a non-colliding motion that unfolds a spherical open chain of length less than 2π , thereby settling the question of reconfigurability and unfoldability of single-vertex origamis. We conclude with some remaining open questions.

There is an asymmetry in the usage of expansive motions in a hemisphere, where the number of pseudo-triangulation induced steps can be bound by $O(n^3)$, a function of n , and the number of belt-shrinking steps, for which the bound depends on the length deficit $2\pi - \alpha$ of the chain. The main remaining question for medium-length open *single-vertex origamis* is:

Open Problem 1 *Design an algorithm based on motions induced by one-degree-of-freedom mechanisms on the sphere (which will necessarily be partially expansive, partially contractive) for which the number of events (when the mechanism's bars align or encounter some similarly easy-to-verify events) can be bounded only in terms of the number n of bars of the chain.*

An algorithmic question remains to be investigated for long open and closed *single-vertex origamis*:

Open Problem 2 *Decide, algorithmically, whether a single vertex-origami whose length exceeds 2π can be reconfigured, without collisions, between two given configurations.*

As a consequence of our work, the study of general *non-colliding* origami folding can now focus on the interaction between panels incident with distinct vertices of the origami pattern, since locally, each single-vertex sub-unit has non-colliding motions. This appears to be a difficult topic, whose systematic investigation is yet to be undertaken.

Acknowledgement. We thank Yang Li for help with the preparation of the figures.

Partial funding for the authors was provided by an NSF International Collaboration grant. The second author was also supported by NSF CCF-0728783 and by a DARPA *Mathematical Challenges* grant, under *Algorithmic Origami and Biology*. All statements, findings or conclusions contained in this publication are those of the authors and do not necessarily reflect the position or policy of the Government.

References

- [1] Devin J. Balkcom and Matthew T. Mason. Introducing robotic origami folding. In *IEEE Internat. Conf. on Robotics and Automation (ICRA)*, pages 3245–3250, April 2004.
- [2] Robert Connelly, Erik D. Demaine, and Günter Rote. Straightening polygonal arcs and convexifying polygonal cycles. *Discrete and Computational Geometry*, 30(2):205–239, 2003. A preliminary version appeared in Proc. 41st Annual Symp. on Found. of Computer Science (FOCS) 2000, pp. 432–442, Redondo Beach, CA.
- [3] Erik D. Demaine and Joseph O’Rourke. *Geometric Folding Algorithms: Linkages, Origami, and Polyhedra*. Cambridge University Press, 2007.
- [4] Thomas C Hull. Rigid origami web site, <http://kahuna.merrimack.edu/~thull/rigid/rigid.html>, 2003.
- [5] Hyun Jin In, Sundeep Kumar, Yang Shao-Horn, and George Barbastathis. Origami fabrication of nanostructured, three-dimensional devices: Electrochemical capacitors with carbon electrodes. *Applied Physics Letters*, 88(8):083104, 2006.
- [6] S. M. Jurga, C. H. Hidrovo, J. Niemczura, Henry I. Smith, and George Barbastathis. Nanostructured origami. In *IEEE Nanotechnology 2003 Proceedings*, nr. TR-4, December 2003.
- [7] Liang Lu and Srinivas Akella. Folding cartons with fixtures: A motion planning approach. *IEEE Trans. on Robotics and Automation*, 16(4):346–356, August 2000.
- [8] Gaiane Panina. Planar pseudo-triangulations, spherical pseudo-tilings and hyperbolic virtual polytopes. Technical Report arXiv:math:MG/0607171, July 2006.
- [9] Günter Rote, Francisco Santos, and Ileana Streinu. Pseudo-triangulations: a survey. In J. E. Goodman, J. Pach, and R. Pollack, editors, *Surveys on Discrete and Computational Geometry - Twenty Years later, Contemporary Mathematics*, vol. 453, pages 343–410. American Mathematical Society, 2008. <http://arxiv.org/abs/math/0612672>.

- [10] Paul W. K. Rothemund. Folding DNA to create nanoscale shapes and patterns. *Nature*, 440(7082):297–302, 2006.
- [11] Harry A. Schmitt, Don E. Waagen, George Barbastathis, and Ileana Streinu. Computational origami for sensor configuration and control. In *Conf. Record of 38th Asilomar Conf. on Signals, Systems and Computers*, vol. 1, pages 811–815. IEEE, 2004.
- [12] Guang Song and Nancy M. Amato. A motion planning approach to folding: From paper craft to protein structure prediction. In *Proc. IEEE Internat. Conference on Robotics and Automation (ICRA)*, vol. 1, pages 948–953. IEEE, 2001.
- [13] Ileana Streinu. Pseudo-triangulations, rigidity and motion planning. *Discrete and Computational Geometry*, 34(4):587–635, 2005. A preliminary version appeared as *A combinatorial approach to planar non-colliding robot arm motion planning*, in Proc. 41st Annual Symp. on Found. of Computer Science (FOCS) 2000, pp. 443–453, Redondo Beach, CA.
- [14] Ileana Streinu and Walter Whiteley. Single-vertex origami and spherical expansive motions. In J. Akiyama and M. Kano, eds., *Proc. Japan Conf. Discrete and Computational Geometry (JCDCG 2004)*, *Lecture Notes in Computer Science*, vol. 3742, pages 161–173, Tokai University, Tokyo, 2005. Springer Verlag.

Extremal Configurations of Hinge Structures

Ciprian Borcea and Ileana Streinu

Abstract

We study body-and-hinge and panel-and-hinge chains in R^d , with two marked points: one on the first body, the other on the last. For a general chain, the squared distance between the marked points gives a Morse-Bott function on a torus configuration space. Maximal configurations, when the distance between the two marked points reaches a global maximum, have particularly simple geometrical characterizations. The three-dimensional case is relevant for applications to robotics and molecular structures.

Keywords: extremal configuration, Morse-Bott function, Hessian matrix, hinge structure, maximum reach, revolute-jointed manipulator.

AMS Subject Classification: 53A17

Introduction

This work is an extension of our study of singularities of hinge structures [BS]. We refer to that paper for basic notions and background.

The hinge structures considered here will be body-and-hinge or panel-and-hinge chains in R^d which have a point marked on the first body and a point marked on the last body. In dimension three, these hinge structures would model serial manipulators with revolute joints with a marked base-point and the end-effector as the other marked point¹. Likewise, the panel-and-hinge case may serve as model for “backbone” protein chains [BT, CP, BS]. Extremal configurations will be those where the squared distance function between the two marked points (origin and terminus, or “head” and “tail”) reaches a local maximum or minimum.

¹The geometrical models have no rotational limitations around the joints and no self-collision prohibitions.

Robotics is obviously concerned with extremal reaches of manipulators and the closely related problem of identifying the total workspace of a robot. A *necessary* condition for extremal configurations was recognized and proven in several papers [D, KW, SD, S]. In the words of [S], where the base-point may be chosen arbitrarily, and the end-point is called “hand”, this necessary condition says: “the line of sight from that point to the hand must intersect all turning axes”². However, all critical points with non-zero value for the squared distance function satisfy the condition, and they grow exponentially with the number of hinges.

In this paper, we refine the study of extremal configurations and obtain, in particular, a very simple *necessary and sufficient* characterization of the global maximum. It may be observed here that the approach used to identify the global maximum configuration (which is unique for generic body-and-hinge chains) cannot be fully adapted for the global minimum, although the panel-and-hinge case offers a fair degree of similarity (Theorems 10 and 11). The distinction, we suggest, stems from the possibility to reinterpret the global maximum as a global minimum of a related problem. Once recognized, the criterion for the global maximum can be proven with completely elementary means.

Global Maximum Theorem: *A body-and-hinge chain is in a global maximum configuration if and only if the segment from the origin to the end-point intersects all hinges in their natural order.*

The panel-and-hinge case has sufficient specificity to warrant separate treatment, most particularly in dimension three. Local extrema must be global extrema, but are not unique if not flat.

In the final section we discuss some variations.

1 A Morse-Bott function for chains with two marked points

In this section we consider body-and-hinge chains with $(n + 1)$ bodies and two marked points, one on the first and the other on the last body. The ambient dimension will be d , and we identify the first body with the fixed reference coordinate system $R^d \equiv B_1$, with the marked point at the origin. The point on the last body will be the end-point.

²This incidence of the origin-to-terminus line with the hinges is understood projectively, that is, includes the possibility of parallelism.

The composition of the *end-point map* $e : (S^1)^n \rightarrow R^d$ with the *squared norm function* $R^d \rightarrow R$ gives the *squared distance function* of the end-point to the origin:

$$F : (S^1)^n \rightarrow R, \quad F(\theta) = \langle e(\theta), e(\theta) \rangle \quad (1)$$

We'll use $T_n = (S^1)^n$ as another notation for the n -torus parametrizing the configuration space of our body-and-hinge chain.

The critical points of F are described by:

Proposition 1. *Let $n \geq d$. If zero is a value of F , then all points in $X_0 = F^{-1}(0) \subset T_n$ are critical points of F . The critical points with non-zero critical values are those configurations which have all hinges (projectively) incident with the line connecting the origin to the end-point.*

In the generic case, $F : T_n = (S^1)^n \rightarrow R$ is a Morse-Bott function, which has only isolated critical points for non-zero critical values, while the fiber over zero $X_0 \subset T_n$, when non-empty, will be smooth, of dimension $n - d$.

Proof: The squared norm on R^d has a critical point at the origin, hence all configurations with the two marked points coinciding (i.e. figuratively, when “the head bites the tail”) will be critical for F .

For non-zero critical values, the argument is similar to the one used in [BS] : at a critical configuration, rotating the part of the chain from hinge A_i on, as a rigid piece, must preserve, infinitesimally, the (squared) distance “head-to-tail”, that is: must produce a velocity vector for the end-point orthogonal to the line between the marked points. That requires the (projective) incidence of hinge and line.

In the generic case, the origin will be a regular value of the end-point map, and, when zero is a value of F , it will give a smooth (not necessarily connected), codimension d fiber $X_0 = F^{-1}(0) \subset T_n$. The fact that the remaining critical points are isolated will follow from the examination of the corresponding Hessian. The Bott non-degeneracy condition [G] along (the connected components of) X_0 will be verified at that stage as well.

1.1 Set-up for computing the Hessian matrix

In a given configuration, hinge number i will be determined by the vector t_i which is the perpendicular projection of the origin on the corresponding hinge,

plus the normal direction to the hyperplane formed by the hinge and the origin. One may keep track of global orientations, but for our computations, a local choice of unit normal ν_i will suffice. Thus $(t_i, \nu_i) = (t_i(\theta), \nu_i(\theta))$ ‘encodes’ the i^{th} hinge.

At any critical point, we may assume the $\theta \in (S^1)^n$ labelling of the configuration space introduced by the following rule: the critical point is $\theta = 0$, and the position for arbitrary $\theta = (\theta_1, \dots, \theta_n)$ is obtained by rotating the last body around the last hinge with angle θ_n , then rotating the last two bodies (as a rigid piece) around the last but one hinge with angle θ_{n-1} , and so on until, at last, the whole (rigid) piece thus formed with all the bodies from the second to the last is rotated around the first hinge with angle θ_1 . All rotations, for $i = n, \dots, 1$, are using the sense dictated by a fixed orientation, say $\{\nu_i, t_i\}$ in the vector plane $[\nu_i, t_i]$ they span.

Let $R_i(\omega)$ stand for the *linear operator in R^d* which gives the *rotation* with angle ω around $[\nu_i, t_i]^\perp$, with the orientation fixed as above. Then, $\frac{\partial R_i}{\partial \omega}(\omega)$ is a skew-symmetric operator vanishing on $[\nu_i, t_i]^\perp$, and we put $\frac{\partial R_i}{\partial \omega}(0) = S_i$.

With a second derivation $\frac{\partial^2 R_i}{\partial \omega^2}(\omega) = -P_i R_i(\omega)$ with P_i denoting the orthogonal projection on the 2-subspace $[\nu_i, t_i]$. Thus: $\frac{\partial^2 R_i}{\partial \omega^2}(0) = -P_i$

We let $x \in R^d$ denote the position of the end-point for the critical configuration under investigation. With the parametrization and notation just described, and the abbreviation $R_i(\theta_i) = R_i$, the *end-point function* is:

$$e(\theta) = R_1 R_2 \dots R_n x + R_1 \dots R_{n-2} (I - R_n) t_n + \dots + R_1 (I - R_2) t_2 + (I - R_1) t_1 \quad (2)$$

This gives:

$$e(0) = x, \quad \frac{\partial e}{\partial \theta_i}(0) = S_i(x - t_i)$$

$$\frac{\partial^2 e}{\partial \theta_i^2}(0) = P_i(t_i - x), \quad \frac{\partial^2 e}{\partial \theta_i \partial \theta_j}(0) = S_j S_i(x - t_i), \quad j < i$$

Considering that $S_i t_i = -||t_i||\nu_i$, the resulting entries for the Hessian matrix are:

$$\frac{1}{2} \frac{\partial^2 F}{\partial \theta_i \partial \theta_j}(0) = \langle S_i(x - t_i), S_j(x - t_j) \rangle + \langle S_j S_i(x - t_i), x \rangle =$$

$$\begin{aligned}
& = \langle S_i(x - t_i), S_j(x - t_j) \rangle - \langle S_i(x - t_i), S_j x \rangle = \langle S_i(t_i - x), S_j t_j \rangle = \\
& = \langle S_i(1 - \frac{\langle x, t_i \rangle}{\langle t_i, t_i \rangle}) t_i, S_j t_j \rangle = (1 - \frac{\langle x, t_i \rangle}{\langle t_i, t_i \rangle}) \langle S_i t_i, S_j t_j \rangle = \\
& = (1 - \frac{\langle x, t_i \rangle}{\langle t_i, t_i \rangle}) \|t_i\| \cdot \|t_j\| \langle \nu_i, \nu_j \rangle
\end{aligned}$$

for $j \leq i$.

We retain the result of this computation as:

Proposition 2. *With adequate parametrization, the symmetric $n \times n$ Hessian matrix for the squared end-point distance function F at a critical configuration $\theta = 0$, with end-point at $e(0) = x$, has entries:*

$$\frac{1}{2} \frac{\partial^2 F}{\partial \theta_i \partial \theta_j}(0) = (1 - \frac{\langle x, t_i \rangle}{\langle t_i, t_i \rangle}) \|t_i\| \cdot \|t_j\| \langle \nu_i, \nu_j \rangle, \quad j \leq i \quad (3)$$

and, after the change of basis $e_i \mapsto \frac{1}{\|t_i\|} e_i$, corresponds with the quadratic form given by:

$$h_{ij} = h_{ji} = (1 - \frac{\langle x, t_i \rangle}{\langle t_i, t_i \rangle}) \langle \nu_i, \nu_j \rangle, \quad j \leq i \quad (4)$$

For a non-zero critical value, the coefficients $\alpha_i = \frac{\langle x, t_i \rangle}{\langle t_i, t_i \rangle}$ are obtained geometrically from the intersections of the line through the origin and the end-point x with the hinges, since these intersections are precisely the points $\frac{1}{\alpha_i} x$, $i = 1, \dots, n$.

When F takes the value zero, the Hessian at a critical point in $X_0 = F^{-1}(0)$ is equivalent to the Gram matrix $h_{ij} = \langle \nu_i, \nu_j \rangle$ of the normals, which is semi-positive definite and has, in the generic case, rank d . Thus, the null-space corresponds to the tangent space of X_0 at the critical point under consideration. This is the Bott non-degeneracy property required along the critical manifold X_0 .

1.2 An upper-bound for the number of isolated critical points

An upper-bound for the number of isolated critical points can be obtained from the following *complexification*: $T_n = (S^1)^n$ is complexified to $(P_1(C))^n$ by considering each circle S^1 as the real locus of the corresponding complex conic:

$$P_1(C) \approx \{x \in P_2(C) : x_1^2 + x_2^2 = x_0^2\}$$

$$\frac{x_1}{x_0} = \cos \theta_i, \quad \frac{x_2}{x_0} = \sin \theta_i$$

With some choice of a reference simplex in each hinge $A_i(\theta)$, say $a_1^i(\theta), \dots, a_{d-1}^i(\theta)$, the condition that the ‘head-to-tail’ line meets this hinge becomes

$$\det[a_1^i(\theta) \dots a_{d-1}^i(\theta) e(\theta)] = 0$$

This defines in $(P_1)^n$ a hypersurface of multi-degree $2(d, \dots, d, 1, \dots, 1)$, with positions d up to the i^{th} coordinate.

The intersection of these n hypersurfaces yields, in the complex count, the number:

$$2^n \int_{(P_1)^n} (h_1 + \dots + h_n)(dh_1 + h_2 + \dots + h_n) \dots (dh_1 + \dots + dh_{n-1} + h_n) = 2^n \sum_{k=0}^{n-1} A(n, k) d^k$$

where h_i stands for the class of a point in the i^{th} factor P_1 , and $A(n, k)$ denote Eulerian numbers. Thus, we have:

Proposition 3. *The number of isolated critical points for $F : T_n \rightarrow R$ is bounded by*

$$2^n \sum_{k=0}^{n-1} A(n, k) d^k \tag{5}$$

1.3 A Meyer-Vietoris sequence (or Morse-Bott theory)

The determination of critical configurations and their indices can be used in the following setting: we *assume* a generic body-and-hinge chain, $n \geq 3$ and zero to be a value of F . This ensures the smoothness of the $(n-3)$ -dimensional fiber $X_0 = F^{-1}(0)$ over zero, and, for small enough $\epsilon > 0$, a diffeomorphism $X_0 \times B_\epsilon^3 \approx F^{-1}[0, \epsilon]$, where B_ϵ^3 stands for the 3-dimensional ball of radius ϵ . The $(n-1)$ -dimensional fiber $X_\epsilon = F^{-1}(\epsilon)$ is thereby identified with $X_0 \times S_\epsilon^2$.

We'll use the notation $T_n = (S^1)^n$ for the n -dimensional torus representing the configuration space of our chain with $(n+1)$ panels and n hinges. For small and nearby values $0 < \gamma < \epsilon < \delta$ we put:

$$U = F^{-1}[0, \delta), \quad V = T_n \setminus F^{-1}[0, \gamma]$$

assuming all non-zero critical values of F greater than δ . Thus, the two open sets cover the torus T_n , and we have *homotopy equivalences*:

$$U \sim X_0, \quad U \cap V \sim X_\epsilon = X_0 \times S^2$$

The corresponding Meyer-Vietoris exact sequence gives:

$$\dots \rightarrow H_i(X_0 \times S^2) \rightarrow H_i(X_0) \oplus H_i(V) \rightarrow H_i(T_n) \rightarrow H_{i-1}(X_0 \times S^2) \rightarrow \dots$$

and in particular, we have the relation of Euler-characteristics:

$$e(X_0 \times S^2) + e(T_n) = e(X_0) + e(V) \quad \text{i.e.} \quad e(X_0) = e(V)$$

By Morse theory [M], the last Euler-characteristic can be expressed in terms of critical points as follows: put c_i for the number of critical points of index i ; then:

$$e(V) = \sum_{i=0}^n (-1)^{n-i} c_i$$

Note: The common convention would be to use $-F$ as the Morse function. Since we go with F , the signs are as above.

We obtained:

Proposition 4. *Suppose $n \geq 3$ and let the origin be a regular value of the end-point map for an otherwise generic chain. Then, the $(n - 3)$ -dimensional manifold X_0 parametrizing all configurations with the origin coinciding with the end-point has the Euler number:*

$$e(X_0) = \sum_{i=0}^n (-1)^{n-i} c_i \quad (6)$$

where c_i is the number of critical points of index i for F on $T_n \setminus X_0$.

1.4 The index of the Hessian matrix: panel-and-hinge case

The restriction to panel-and-hinge chains brings new structural aspects. We note first the presence of natural transformations of the configurations space $T_n = (S^1)^n$:

Transformations: Given any configuration, one may consider the hyperplane through one marked point and some hinge, and reflect the part of the chain from point to hinge in this hyperplane. The resulting transformation is obviously its own inverse i.e. an *involution*. Note that we may always reposition the structure with its first panel in its fixed location. Since the composition of reflecting in the first and then last panel gives a global rotation of the chain, these two operations represent the same transformation of T_n and this gives $2n - 1$ involutions on the configuration space, all transforming the fibers of F to themselves. Two such involutions commute when they implicate the same marked point or the respective portions of the chain do not overlap.

We have seen that, in case F reaches 0, $F^{-1}(0)$ is part of the critical locus, but all critical points for non-zero critical values are isolated in the generic case. The following definition refers to these isolated critical configurations corresponding to non-zero critical values of F :

Definition 5. *The 2^n flattened configurations when all panels lie in the same hyperplane (i.e. the codimension one subspace of the first panel, which is identified here with $x_d = 0$ in R^d) will be called flat critical configurations (points), while critical configurations which do not have all panels in the same hyperplane will be called non-flat critical configurations (points).*

It will be observed that, in a non-flat critical configuration, a new panel normal direction requires two consecutive hinges to meet the line from the origin to the

end-point in the same point and we'll call such a point a fold point (on the "head-to-tail" line). Fold points will be considered as labelled by the corresponding pair of consecutive hinges and ordered via this labelling.

Note: Flat critical configurations are fixed points of all involutions described above and the group they generate, while non-flat critical configurations have an orbit of cardinality 2^c under this group, where c is the number of fold points on the "head-to-tail" line. Assuming a generic case, $c + 1$ will give the number of distinct hyperplanes determined by origin and hinges .

We settle first the case of all flat critical configurations. Then all normal directions are the same and (regardless of the choice $\pm n_i$) Proposition 2 gives the signature as that of the (quadratic form with) matrix:

$$h_{ij} = h_{ji} = 1 - \frac{\langle x, t_i \rangle}{\langle t_i, t_i \rangle} = \beta_i, \quad j \leq i \quad (7)$$

Lemma 6. *The signature of the above matrix is determined by the signs of:*

$$\beta_1 - \beta_2, \beta_2 - \beta_3, \dots, \beta_{n-1} - \beta_n, \beta_n$$

Proof: The symmetric matrix $H = (h_{ij})$ is the expression of the quadratic form $q(x) = x^t H x$ in the standard basis $e_i, i = 1, \dots, n$. If we do the (unimodular) change of basis:

$$e_i \mapsto \tilde{e}_i = e_i - e_{i+1}, \quad i = 1, \dots, n-1; \quad e_n = \tilde{e}_n$$

we obtain a diagonal matrix with the indicated entries.

As mentioned in Proposition 2, if we mark the end-point by x , and the intersections of the "head-to-tail" line with the k^{th} hinge is $a_k x$, then $\beta_k = 1 - a_k^{-1}$. For a generic chain in a flattened position, hinges will intersect the "head-to-tail" line in different points, and we'll have a non-degenerate Hessian. Since by definition, the index of a symmetric matrix is the number of negative eigen-values, we have:

Proposition 7. *The index of a flat critical configuration with hinges meeting the line from the origin to the end-point x at $a_k x$, $k = 1, \dots, n$, is the number of negative values in the list:*

$$1 - \frac{1}{a_n}, \frac{1}{a_n} - \frac{1}{a_{n-1}}, \dots, \frac{1}{a_2} - \frac{1}{a_1} \quad (8)$$

Corollary 8. *A flat critical configuration is a local maximum if and only if the hinges meet the oriented segment $(0, x)$ in points $a_k x$, $k = 1, \dots, n$, lined-up in their natural order, that is:*

$$0 < a_1 < a_2 < \dots < a_n < 1 \quad (9)$$

For a convenient geometric formulation of the existence of a local minimum in a flat configuration we conceive of the “head-to-tail” line as completed to a projective line, and the complement of the affine segment $[0, x]$ gives then the open arc from 0 to x passing through the “point at infinity”.

Corollary 9. *A flat critical configuration is a local minimum if and only if the hinges meet the oriented arc from 0 to x passing through the point at infinity in their natural order. This means one the following:*

$$a_n < \dots < a_2 < a_1 < 0 \quad \text{or} \quad (10)$$

$$a_k < \dots < a_1 < 0 < 1 < a_n < \dots < a_{k+1} \quad \text{or} \quad (11)$$

$$1 < a_n < \dots < a_2 < a_1 \quad (12)$$

Remark: In fact, the projective formulation allows some relaxation in the genericity assumptions and one of the hinges may be parallel to the “head-to-tail” line in that flat configuration.

1.5 Extremal configurations

In this section we refine our description of extremal configurations of panel-and-hinge chains in R^d with two marked points.

We’ll elaborate on our Corollaries 8 and 9 and address possible maxima and minima at non-flat critical configurations. One should remain aware of the involutions described in subsection 1.3. Recall that a fold point on the “head-to-tail” line is common to two consecutive hinges, say $a_k x = a_{k+1} x$. When speaking of the ordering of intersections of hinges with the “head-to-tail” line, either ordering may be envisaged for the two hinges, but we intend the ordering requested in the statement. The function F is the squared distance from “head” to “tail”.

Theorem 10. *A local maximal configuration for F is characterized by the fact that all hinges intersect the oriented segment $(0, x)$ in the natural order. Moreover, a local maximum is in fact the global maximum and is unique modulo the natural transformations generated by the involutions described above. Thus there are 2^μ maximal configurations equivalent under natural transformations, where μ is the number of fold points for any and all of them.*

Proof: Let us call *flat subsystem* in a non-flat critical configuration the panel-and-hinge structure obtained by retaining the consecutive axes which lie in the same hyperplane through the “head-to-tail” line i.e. the hinges corresponding to a specific normal direction ν_i (up to the first occurrence of a fold point, or between a fold point and the next, or up from the last fold point). The “head” and “tail” are inherited from the full configuration. Then the expression (4) we obtained for the Hessian shows that a local maximum requires all flat subsystems to be flat local maxima. Thus, by Corollary 8, all hinges intersect the oriented segment $(0, x)$ in the natural sequential order.

As a consequence, we may trace a “red line” on consecutive panels by following the segment $[0, x]$ in our local maximum configuration. Thus, any other configuration will display the “red line” as a polygonal arc from “head” to “tail” proving that our local maximum is the global maximum. It also follows that any other local maximum must have exactly the same pattern of planar subsystems and therefore be obtained from our maximal configuration by some composition of involutions.

Theorem 11. *A local minimal configuration for a non-zero value of F is characterized by the fact that all hinges intersect the oriented projective arc from 0 to x passing through the point at infinity (i.e. the complement of $[0, x]$) in the natural order. Moreover, if such a local minimum exists, it is in fact the global minimum and 0 is not a possible value for F . When 0 is not a possible value for F , all minima are achieved at 2^ν minimal configurations equivalent under natural transformations, where ν is the number of fold points for any and all of them.*

Proof: Simple adaptation of the “red line” argument presented above.

2 The global maximum as a global minimum

The criteria obtained above for extremal configurations of marked panel-and-hinge chains in R^d offer obvious suggestions for the more general *body-and-hinge case*.

It is an immediate observation that a marked body-and-hinge chain for which the segment from the origin to the end-point intersects all hinges in their natural order is in a global maximum configuration, since for any other configuration, the previous segment (drawn as a “red segment”) becomes a polygonal arc longer than the new segment from the origin to the end-point. What is less immediate is that any body-and-hinge chain actually reaches its global maximum in such a configuration. This will be proven by relating the global maximum to a global minimum.

Theorem 12. *Let a body-and-hinge chain be presented in a fixed configuration, with the origin as the marked point of the first panel, e the end-point (on the last body) and hinges given by codimension two affine subspaces A_i , $i = 1, \dots, n$. Consider variable points on each hinge $a_i \in A_i$ and the length of the polygonal arc going from the origin to the end-point through the points a_i in their natural order:*

$$f(a_1, \dots, a_n) = \|a_1\| + \|a_2 - a_1\| + \dots + \|a_n - a_{n-1}\| + \|e - a_n\| \quad (13)$$

The global maximum distance between the marked points of the given body-and-hinge chain equals the global minimum of the function f .

Proof: Note that f is continuous and a global minimum always exists.

The proof will follow from the simple case of a single hinge ($n=1$). In this case, a_1 must be the unique point of A_1 which allows a rotation of the segment from a_1 to e around A_1 to become a continuation of the segment from the origin to a_1 . Repeating this observation with respect to a_{i-1}, a_{i+1} and the hinge A_i , shows that the chain can be reconfigured so that the polygonal arc realizing the global minimum becomes a straight segment from the origin to the end-point intersecting all hinges in their natural order. That is the global maximum.

We have proven at the same time our:

Global Maximum Theorem: *A body-and-hinge chain is in a global maximum configuration if and only if the segment from the origin to the end-point intersects all hinges in their natural order. For a generic body-and-hinge chain, this global maximum is unique.*

Remarks: (i) In general, a body-and-hinge chain may well have local maxima which are not global maxima. By (4), the segment from the origin to the end-point must intersect all hinges.

(ii) f is continuous, but not differentiable when $a_i = a_{i+1}$ for some i .

Examples show that global minima for body-and-hinge chains do not have to respect the pattern observed for panel-and-hinge chains. We may consider, for instance, just two hinges (say in R^3) and take the second body (containing these hinges) as reference. Then the marked points trace circles, each around a hinge. Examples of global minima with the hinges intersecting in the order A_2, A_1 the projective arc from origin to the endpoint passing thorough infinity, can be easily produced. What still holds true, by (4), for any local minimum is the fact that all hinges intersect the projective arc from the origin to the end-point, passing through the point at infinity.

3 Variations on the same theme

A few ‘variations’ of these techniques should be mentioned before concluding. Again, the issues are important in robotics and the necessary conditions have been detected in the literature [SR, SD].

If we abandon the first marked point, we’ll rather be concerned with the squared distance from the end-point to the first hinge. The resulting critical configurations for non-zero values will require the *perpendicular* from the end-point to the first hinge to meet (projectively) all hinges. The global maximum will require the natural ordering of these intersections on the segment from the foot of the perpendicular to the end-point.

In dimension three, a similar scenario holds if we abandon both marked points and only inquire about the squared distance between the first hinge and the last hinge. Critical configurations for non-zero values will have the common perpendicular of these two hinges intersect (projectively) all the intermediate hinges. The global maximum will require the natural ordering.

References

- [BS] Borcea, C.S. and Streinu, I.: *Singularities of Hinge Structures*, Effective Methods in Algebraic Geometry, MEGA 2005. Posted on arXiv, Dec. 2008.
- [BT] Braden, C. and Tooze, J.: *Introduction to Protein Structure*, 2nd edition, Garland Publishing, Inc., New York (1998).

- [CP] Canny, J. and Parsons, D.: *Geometric problems in molecular biology and robotics*, Proceedings of the Second International Conference on Intelligent Systems for Molecular Biology, Stanford, CA, 1994.
- [D] Derby, S.: *The maximum reach of revolute jointed manipulators*, Mechanism and Machine Theory, Vol. **16**, Issue 3 (1981), 255-261.
- [G] Guest, M.: *Morse theory in the 1990's*, in *Invitations to geometry and topology*, Oxford Grad. Texts Math. **7**, Oxford Univ. Press (2002), 146-207.
- [KW] Kumar, A. and Waldron, K.J.: The workspaces of a mechanical manipulator, ASME Jour. Mech. Design, Vol. **103**, Issue 3 (1981), 665-672.
- [M] Milnor, J.: *Morse Theory*, Annals of Mathematics Studies, vol. **51**, Princeton University Press, 1963.
- [S] Selfridge, R.G.: *The reachable workarea of a manipulator*, Mechanism and Machine Theory, Vol. **18**, Issue 2 (1983), 131-137.
- [Se] Selig, J.M.: *Geometric Methods in Robotics*, Springer-Verlag, New York (1996).
- [SR] Shimano, B. and Roth, B.: *Dimensional synthesis of manipulators*, Third CIMS-IFToMM Int. Symp. Theory and Practice of Robots and Manipulators, Udine, Italy (1978).
- [SD] Sugimoto, K. and Duffy, J.: *Determination of extreme distances of a robot hand -Part 1: A general theory*, ASME Jour. Mech. Design, Vol. **103**, Issue 3 (1981), 631-636.

Pseudo-Triangulations, Rigidity and Motion Planning

Ileana Streinu *

Abstract

This paper proposes a combinatorial approach to planning non-colliding trajectories for a polygonal bar-and-joint framework with n vertices. It is based on a new class of simple motions induced by expansive one-degree-of-freedom mechanisms, which guarantee non-collisions by moving all points away from each other. Their combinatorial structure is captured by *pointed pseudo-triangulations*, a class of embedded planar graphs for which we give several equivalent characterizations and exhibit rich rigidity theoretic properties.

The main application is an efficient algorithm for the *Carpenter's Rule Problem*: convexify a simple bar-and-joint planar polygonal linkage using only non-self-intersecting planar motions. A step of the algorithm consists in moving a pseudo-triangulation-based mechanism along its unique trajectory in configuration space until two adjacent edges align. At the alignment event, a local alteration restores the pseudo-triangulation. The motion continues for $O(n^3)$ steps until all the points are in convex position.

1 Introduction

We present a combinatorial solution to the *Carpenter's Rule Problem*: how to plan non-colliding reconfigurations of a planar robot arm. The main result is an efficient algorithm for the problem of continuously moving a simple planar polygon to any other configuration with the same edge-lengths and orientation, while remaining in the plane and never creating self-intersections along the way. This is done by first finding motions that convexify both configurations with expansive motions (which never bring two points closer together) and then taking one path in reverse.

All of the constructions are elementary and are based on a novel class of planar embedded graphs called *pointed pseudo-triangulations*, for which we prove a variety of combinatorial and rigidity theoretical properties. More prominently, a pointed pseudo-triangulation with a removed convex hull edge is a one-degree-of-freedom expansive mechanism. If its edges are seen as rigid bars (maintaining their lengths) and are allowed to rotate freely around the vertices (joints), the mechanism follows (for a well defined, finite time interval) a continuous trajectory along which no distance between a pair of points ever decreases. The expansive motion induced by these mechanisms provide the building blocks of our algorithm.

Historical remark. This paper is a systematic, detailed and self-contained presentation of a 10-page conference version [73] which appeared in 2000. In the same conference appeared the independent solution of [30] for the Carpenter's Rule problem, which has since been published as a full paper. Therefore, all the references we give to *previous work* refer to the state of affairs in 2000. The pointed pseudo-triangulations and their special combinatorial and rigidity-theoretical

*Department of Computer Science, Smith College, Northampton, MA 01063, USA. streinu@cs.smith.edu.
This work was partially supported by NSF grant CCR-0105507.

properties, which are the highlight of our solution, found a life of their own since 2000, and a flurry of papers (some extending the results presented here) emerged. For completeness, we also include in the Conclusions a list of references to these recent results.

In the remaining of the introduction, we give an informal high-level preview of the results and their connection with previous work. Precise definitions and complete proofs are given in the rest of the paper. Section 2 contains the definition of pointed pseudo-triangulations and several of their combinatorial characterizations. Sections 3 and 4 prove the rigidity theoretic properties of pointed pseudo-triangulations on which the whole approach relies. Section 5 describes a few simple algorithms for computing pointed pseudo-triangulations of planar point sets and polygons. Section 6 gives the description of the global convexification motion and the complexity analysis of the combinatorial, non-algebraic part of the algorithm. We conclude with some suggestions for further research.

Frameworks and Robot arms. A *bar-and-joint framework* is a graph $G = (V, E)$ embedded in the plane with rigid bars corresponding to the edges (the edge lengths are considered given and fixed). The bars are free to move in the plane around their adjacent joints (vertices), as long as their lengths are preserved. The motions impose no restriction on the non-edges, which may increase or decrease freely. In this general model, edges may cross and slide over each other during the motion, but in this paper we are interested in avoiding collisions and will not allow this. Of particular interest are the *expansive motions*, where no pairwise interdistance between vertices ever decreases during the motion, thus guaranteeing non-collision.

A *chain*, *linkage* or *robot arm* is a planar framework whose underlying graph is a simple (non-self-intersecting) path, and a *closed chain* is a simple planar polygon. Throughout, n will denote the number of its vertices. *Straightening a chain* refers to moving it continuously until all its vertices lie on one line with non-overlapping edges. *Convexifying a closed chain* means moving it to a position where it forms a simple convex polygon. Other types of frameworks of interest in this paper include Laman graphs and pointed pseudo-triangulations, defined in Section 2.1.

The Carpenter’s Rule Problem: Given a closed chain, orient it so that the interior lies to the left when walking along the polygon in the positive direction. While avoiding self-collisions and staying in the plane, we want to continuously reconfigure the chain from an initial to a final configuration with the same orientation. It suffices to show that we can convexify the chain. Then, to move between any two similarly oriented configurations, we will take one path in reverse. Indeed, it is easy to move between two distinct convex positions, see [9]. The Carpenter’s Rule Problem asks: *Is it always possible to straighten a planar linkage, or to convexify a planar chain?*

This question has been open since the 1970’s. Recently, Connelly, Demaine and Rote [30] have answered it in the affirmative. Their solution still left open the problem: *Find, algorithmically, a finite sequence of simple (finitely described) motions to straighten a linkage, or to convexify a polygon.*

Previous Results on Reconfiguring Linkages. The general techniques for solving motion planning problems based on roadmaps [70, 27, 16, 17] work well on problems with constant number of degrees of freedom, but would yield exponential algorithms in our case. Under various conditions, problems about reconfiguration of linkages range in complexity from polynomial [52] to NP- and even PSPACE-hard, see [43, 84, 44].

The particular problem of straightening bar-and-joint linkages and convexifying polygons has accumulated a distinguished history, with some approaches going back to a question of Erdős

[34]. See Toussaint [79] for a fascinating account. There are abundant connections with work done in the computational biology, chemistry and physics literature and motivated by topics such as protein folding or molecular modeling. When crossings are allowed, Lenhart and Whitesides [52] have shown that the configuration space has at most two connected components and gave a linear algorithm for convexification based on simple motions which displace (relatively) only a constant number of joints at a time. Recent results in the mathematics literature [45, 58, 80] aim at understanding the topology of the configuration space of closed chains, but they allow crossings.

Studying reconfigurations of linkages with non crossing motions has received a recent impetus in [54], and results on planar linkages using spatial motions [19, 11], trees, three- and higher-dimensional linkages [20, 29] have followed. The Carpenter’s Rule question, raised in the 1970’s in the Topology community by G. Bergman, U. Grenander, S. Schanuel (cf. [48]) and independently in the early 1990’s in the Computer Science community by two groups (W. Lenhart and S. Whitesides, resp. J. Mitchell), seems to have first appeared in print in [52] and [48]. It was recently settled by Connelly, Demaine and Rote [30]: all chains can be convexified, all linkages can be straightened. Their approach is based on Rote’s seminal idea of using *expansive motions* to guarantee non-collisions. They first prove (using linear programming duality and Maxwell’s theorem, specifically a technique originating in Crapo and Whiteley [33] and Whiteley [81]) that there always exists an infinitesimal expansive motion, i.e. one which never decreases any distances. The actual velocities can be found using linear programming. Then they provide a global argument, showing the existence of a continuous deformation obtained by integrating the resulting vector field.

The Main Results. We strengthen and provide an algorithmic extension of the above mentioned Carpenter’s Rule result of [30]. We show how to compute a path in configuration space, consisting in at most $\mathcal{O}(n^3)$ simple motions along algebraic curve segments, between any two polygon configurations. Along the way, we obtain a result of independent interest in Rigidity Theory. Namely, we characterize a family of planar infinitesimally rigid, self-stress-free frameworks called *pointed pseudo-triangulations*, which yield one-degree-of-freedom (1dof) *expansive* mechanisms when a convex hull edge is removed.

Overview of the Convexification Algorithm. The convexifying path, seen as the collection of the $2n$ trajectories of the $2n$ coordinates $(x_i, y_i), i = 1, \dots, n$ of the vertices of the polygon, is a finite sequence of algebraic curve segments (arcs) connecting continuously at their endpoints.

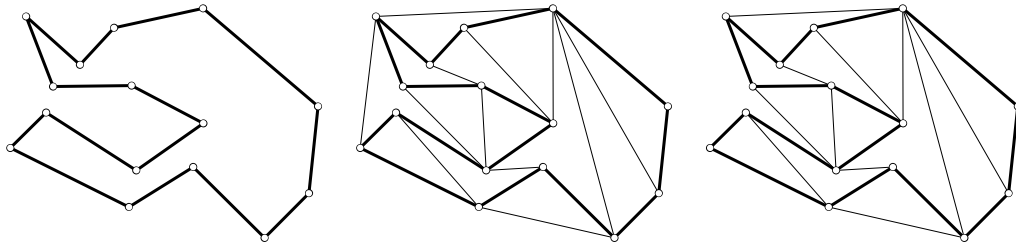


Figure 1: Left: A simple polygon. Middle: one of its pointed pseudo-triangulations. Right: the pointed pseudo-triangulation mechanism obtained by removing a convex hull edge.

Each arc corresponds to the unique free motion of the expansive, one-degree-of-freedom (1dof) mechanism induced by a planar pointed pseudo-triangulation of the given polygon, where a convex hull edge has been removed and a remaining edge has been pinned down. The mechanism is constructed by adding $n - 4$ bars to the original polygon in such a way that there are no crossings, each vertex is incident to an angle larger than π and exactly one convex hull edge is missing. See Fig. 1. We show that this can be done algorithmically in $\mathcal{O}(n)$ time. The mechanism is then set in motion by pinning down one edge and rotating another edge around one of its joints. The framework now moves *expansively*, thus guaranteeing a collision-free trajectory. One step of the convexification algorithm consists in moving this mechanism until two incident edges align, at which moment it ceases to be a pointed pseudo-triangulation. We either freeze a joint (if the aligned edges belong to the polygon) and locally patch a pointed pseudo-triangulation for a polygon with one less vertex, or otherwise perform a local flip of the added diagonals. See Fig. 2.

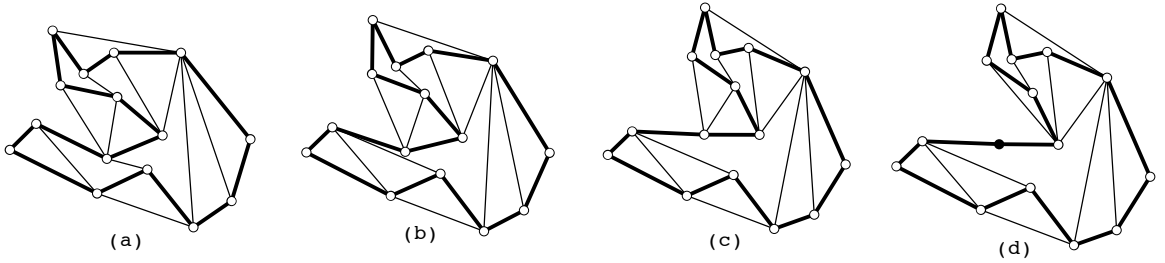


Figure 2: (a) A pointed pseudo-triangulation mechanism just before an alignment event and (b) after the event, when a flip was performed. (c) Continuing the motion, the next event aligns two polygon edges. (d) The aligned vertex (black) is frozen, the pseudo-triangulation is locally restructured and the motion can continue.

There are many ways to construct the initial pointed pseudo-triangulation or to readjust it at an alignment event. For the sake of the analysis, we use a canonical pseudo-triangulation based on shortest-path trees inside the polygon and its pockets. During the convexification process, the total number of bends in the shortest paths, which is bounded by $\mathcal{O}(n^2)$ for n active (not frozen) vertices, decreases by at least one at each flip event. There are at most $\mathcal{O}(n)$ freeze-events (more precisely, as many as there are reflex vertices). The total number of events, and thus the number of steps induced by simple pseudo-triangulation mechanism motions, is therefore $\mathcal{O}(n^3)$.

Formally stating and proving these results will take the rest of the paper. It will require background material in Rigidity Theory and some intuitions from Oriented Matroids and Visibility. In the context of these fields, we offer next an overview of the theoretical contributions of this paper.

Pointed pseudo-triangulations in the context of Oriented Matroids and Rigidity Theory. Our approach is based on the idea of abstracting (partial) oriented-matroidal properties that hold throughout a portion of a continuous motion. To the best of our knowledge, this idea has not been used before in any other context. This leads to our introduction of the *pointed pseudo-triangulations* of planar point sets (and their further generalization in the

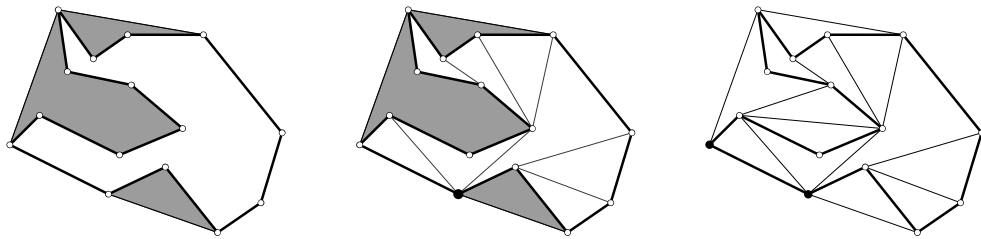


Figure 3: Left: The pockets of the polygon from Fig. 1. Middle: the shortest-path-tree pseudo-triangulation of the interior of the polygon. The source vertex of the tree is black. Right: a complete pointed pseudo-triangulation obtained by taking shortest path trees in all the pockets, and in the interior of the polygon.

companion paper [74] to rank 3 affine oriented matroids, via planar pseudo-point configurations). We emphasize that for these objects, the focus is on *pointedness* (incidence with an angle larger than π), rather than on the partitioning into pseudo-triangular faces (which appears as a consequence), and that they have special properties not shared (and not considered) by other types of pseudo-triangulations previously defined in the literature. Indeed, the combinatorial and rigidity theoretic properties of pointed pseudo-triangulations, as well as their application to expansive mechanisms for finding non-colliding paths in configuration spaces, were discovered by the author and have first appeared in the original conference paper [73], of which the current paper is a comprehensive development.

At the technical, rigidity theoretic level, we prove a new result on the non-existence of self-stresses in bar-and-strut pointed and non-crossing frameworks, which are more complex structures than those arising from polygons. We also prove that pointed pseudo-triangulations are *always* self-stress-free, and therefore (because of having the right number $2n - 3$ of edges) infinitesimally rigid. To the best of our knowledge, no other combinatorially-defined class of frameworks was previously shown to possess such strong rigidity theoretical properties. As a consequence, we obtain a generalization of a key lemma in [30] regarding the existence of expansive motions of certain families of linkages, in the strongest way in which this can be done *combinatorially*.

All these preliminary results imply that the configuration space of 1dof pseudo-triangulation frameworks is *smooth* in the neighborhood of a given realization, as long as the combinatorial structure of the embedding (what we call the *combinatorial pseudo-triangulation*) doesn't change. In particular, a generic planar one-degree-of-freedom mechanism (Laman graph with a convex hull removed edge), when embedded as a pointed pseudo-triangulation, always *moves* along a unique, well-defined, one-dimensional trajectory. To the best of our knowledge, this is the first result in the Rigidity Theory literature where the smooth nature of the configuration space is characterized *combinatorially*: in this case, the combinatorics consists not only in the underlying graph structure (a planar Laman graph), but also in the *oriented matroid* properties of the underlying point set on which the Laman graph is embedded (the pointedness of the embedding). Indeed, to contrast the general situation to what we exhibit here and help the reader understand this point, one can easily exhibit examples of infinitesimally flexible frame-

works which are *rigid*. In these cases, the infinitesimal motion doesn't extend to a finite motion, and locally the configuration space is an isolated point. Or, one can exhibit examples where the framework is flexible, but lies at a singular point in its configuration space, which complicates the design of which trajectory to follow. Our results imply that this can *never* happen for pointed pseudo-triangulation mechanisms.

Expansive Motions and Pseudo-triangulations in the context of previous work. Our pointed pseudo-triangulations are slight specializations (from smooth obstacles to simple point sets, from emphasis on pseudo-triangular faces to emphasis on pointedness of vertices) of those introduced by Pocchiola and Vegter [62, 63] in their study of the visibility complex and recently applied to kinetic geometric algorithms [1, 15].

A 1dof infinitesimally expansive mechanism obtained from a pointed pseudo-triangulation is a combinatorial abstraction and a canonical representation of one of the many basic feasible solutions of infinitesimally expansive motions, that the linear programming approach of [30] would find for a certain position of the polygon in its configuration space. This idea is further developed in [68].

We characterize pointed pseudo-triangulations in several equivalent ways. Some of these are specialized versions of Laman's $2n - 3$ count and Henneberg constructions from combinatorial rigidity [82, 38]. The proof of correctness of our approach derives from these properties, as well as from a generalization, from simple polygons to the wider class of pointed pseudo-triangulation frameworks, of the approach used in [30] based on Linear Programming duality and Maxwell's theorem. This generalization also simplifies the argument needed to guarantee the existence of a global motion, which is now a simple consequence of a basic, fundamental theorem in Calculus, or, alternatively, of the most fundamental theorem in the theory of ordinary differential equations.

References. For background terminology and basic results in rigidity theory, we refer the reader to [66, 82, 83, 38]. In particular, rigidity, infinitesimal (first-order) and generic rigidity, as well as classical results on 2-dimensional rigidity such as Laman's theorem, the Henneberg constructions and Maxwell's Theorem are to be found there. For oriented matroids, see [21], although we won't need more than the intuitions gained through familiarity with the local sequences of [37] (known also as hyperline sequences in [22]), see also [72, 23].

Notation, abbreviations and terminology. Our setting is the Euclidian plane. We abbreviate "counter-clockwise" as *ccw* and "one-degree-of-freedom mechanism" as *1dof mechanism*.

For self-containment, we introduce all the basic terminology and definitions from rigidity theory, graph theory and oriented matroids. Other concepts used in this paper, such as pointed and minimum pseudo-triangulations, combinatorial frameworks, semi-simplicity and expansive 1dof mechanisms are (to the best of our knowledge) new and have not been defined elsewhere. In contrast with the preliminary version of this paper [73], we have settled for a friendlier terminology, and use *pointed* instead of *acyclic* set of vectors, *pointed pseudo-triangulation* instead of *acyclic* or minimum pseudo-triangulation and *expansive* instead of *monotone* motion or mechanism. See the *Notes* in the Conclusion section for further remarks on the choice of terminology.

2 Combinatorial Properties of Pointed Pseudo-Triangulations

We start by defining *pointed pseudo-triangulations* and derive their main combinatorial properties. At the end of this section, we would have acquired the first piece of evidence that pointed pseudo-triangulations have relevant rigidity theoretic properties: their underlying graphs are minimally generically rigid (Laman) graphs. For this to become a useful algorithmic tool, we'll have to prove later that the rigidity property holds for *any* pseudo-triangular embedding, not just generically.

2.1 Definitions: Pointed Pseudo-Triangulations

Throughout the paper, $G = (V, E)$ will denote a graph with $n = |V|$ vertices and $m = |E|$ edges. The vertex set will be taken as $V = [n] := \{1, 2, \dots, n\}$.

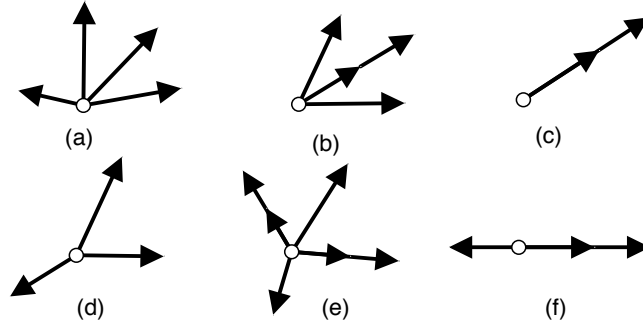


Figure 4: (a), (b) and (c) Pointed and (d), (e) and (f) non-pointed sets of vectors.

Graph Embedding and Planarity. An *embedding* or *drawing* $G(\mathcal{P})$ of the graph G on a set of points $\mathcal{P} = \{p_1, \dots, p_n\} \subset \mathbb{R}^2$ is a mapping of the vertices V to points in the Euclidean plane $i \mapsto p_i \in \mathcal{P}$. The edges $ij \in E$ are mapped to straight line segments $\overline{p_i p_j}$. The embedding $G(\mathcal{P})$ is *planar* if distinct endpoints of edges are mapped to distinct points and edges are mapped to disjoint line segments, except when the edges are incident (in which case, their corresponding segments are allowed to have only one point in common): $\overline{p_i p_j} \cap \overline{p_k p_l} = \emptyset$, for any pair of non-adjacent edges $ij, kl \in E, i, j \notin \{k, l\}$, and $\overline{p_i p_j} \cap \overline{p_i p_k} = \{p_i\}$, for $j \neq k$. A graph G is *planar* if it has a planar embedding.

Pointed Graph Embedding. A set of vectors in $\mathbb{R}^2$ (with a common origin) is *pointed* if it is strictly contained in a half-plane, and *non-pointed* otherwise. Equivalently, some consecutive pair of vectors (in the circular ccw order around the common vertex) spans a *reflex* angle (larger than π). See Fig. 4. Algebraically, this is expressed as the non-existence of a linear combination with all positive coefficients, summing them to zero. A *pointed graph* is an embedded graph such that the edge vectors around each vertex are a pointed set. See Fig. 5.

Simple Polygon. A simple polygon is a *planar embedding* of a cycle graph. In this case, there is a well-defined and connected interior and exterior of the polygon. We will assume that the

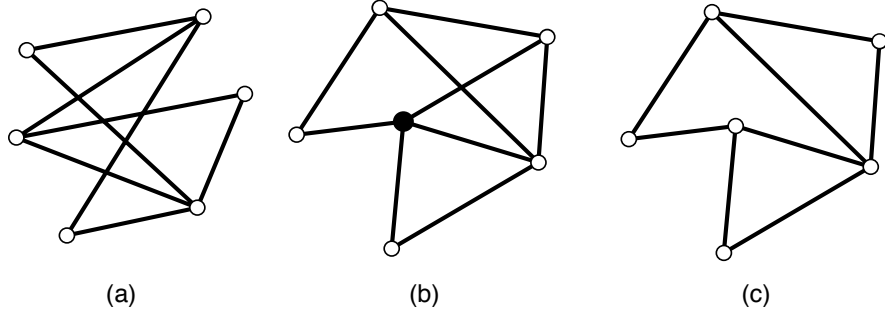


Figure 5: (a) A pointed graph embedding. (b) A non-pointed graph, due to the black non-pointed vertex. (c) A pointed and planar graph.

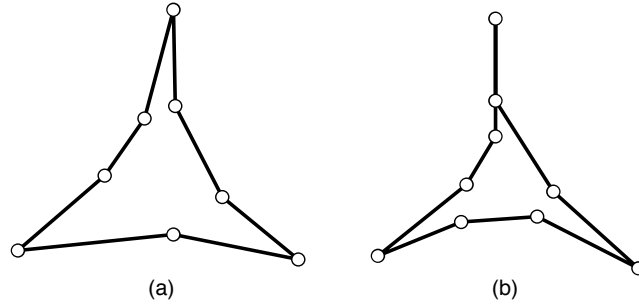


Figure 6: (a) A pseudo-triangle. (b) A semi-simple pseudo-triangle.

labelling of the vertices as $\{1, \dots, n\}$ is in ccw order, i.e. such that the interior lies to the left when the boundary of the polygon is traversed in increasing order of its labels. *We will work only with simple polygons having no angle equal to π , which are therefore **pointed**.* The case when a vertex i is *aligned* (incident to an angle of π) is reduced to the pointed case by the operation of *freezing* the aligned vertex: i is eliminated and its two incident edges $(i-1)i$ and $i(i+1)$ are replaced by a single edge joining the vertices $i-1$ and $i+1$. Throughout, index arithmetic is done mod n in the set $[n] := \{1, \dots, n\}$.

Pseudo-triangle. A *pseudo-triangle* is a simple (pointed) polygon with exactly three convex vertices, called *corners*. The three corners are on the convex hull of the pseudo-triangle and are joined by three inward convex polygonal chains. In particular, a triangle is a pseudo-triangle. See Fig. 6(a). We also introduce *semi-simple pseudo-triangles* as a special case which allows for some degeneracies: some of the inner convex angles (the corners) may be zero, but none of the inner reflex angles is allowed to be π or 2π . Moreover, we do not allow the two overlapping edges of a zero-angle corner to completely coincide: their other endpoints must be different. See

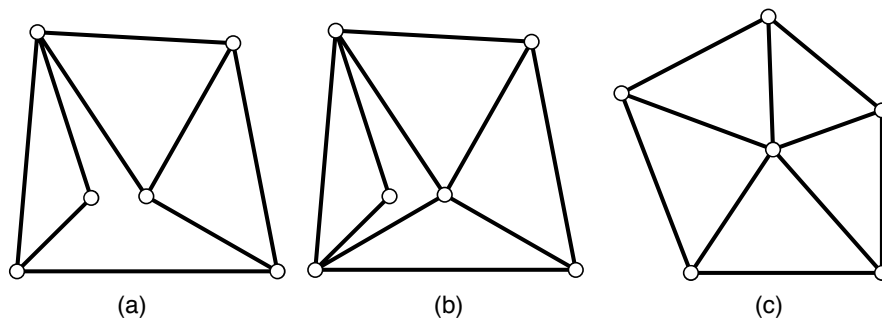


Figure 7: (a) A minimum, pointed pseudo-triangulation. (b) A non-minimum non-pointed pseudo-triangulation which contains a minimum one. (c) A non-minimum non-pointed pseudo-triangulation which does not contain a minimum pseudo-triangulation.

Fig. 6(b).

Pseudo-Triangulation. A pseudo-triangulation is a planar graph embedding whose outer face is convex and all interior faces are pseudo-triangles. A *minimum pseudo-triangulation* has the least number of edges among all pseudo-triangulations on the same point set. A *pointed pseudo-triangulation* is pointed, as an embedded graph. These two definitions will turn out to be equivalent (Theorem 2.3). See Fig. 7.

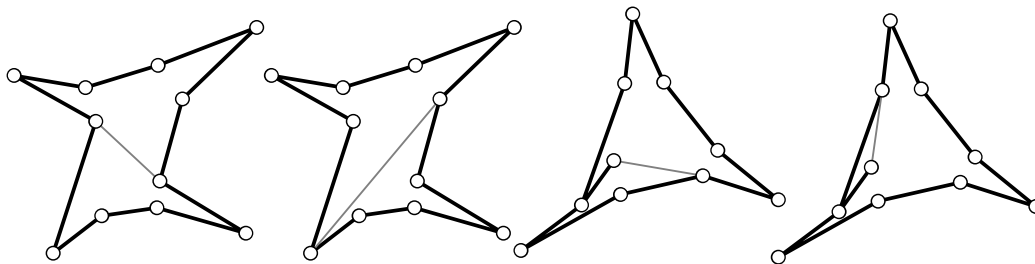


Figure 8: (a) A pseudo-quadrilateral and its two possible subdivisions into two pseudo-triangles, illustrating a diagonal *flip*. (b) A degenerate pseudo-quadrilateral and the associated flip.

Diagonal Flips in Pseudo-Quadrilaterals. More generally, if we focus on the convex vertices of a simple polygon (and call them, for consistency, *corners*) and on the inner convex chains between them, we may refer to the polygon as being a *pseudo- k -gon* if it has exactly k corners. Fig. 8 shows examples of pseudo-quadrilaterals ($k = 4$).

A *tangent* to a convex chain is a line segment with one endpoint on the chain, and whose supporting line contains the chain on one side (doesn't cut through it). A *bitangent* to two convex chains is a line segment whose endpoints are each tangent to one of the chains. Two

pseudo-triangles sharing an edge are merged into a pseudo-quadrilateral when the common edge is removed. If the removed edge was incident to a vertex of degree two, the resulting face will have a *dangling* edge (incident to a vertex of degree one). We still can treat it as if it was a sort of degenerate pseudo-quadrilateral: indeed, seen from inside the face, it has exactly four corners. The dangling edge is *doubled* (traversed twice) as we walk around the boundary of the face. Notice that in this case the pseudo-quadrilateral face has an interior angle equal to 2π (at the degree-one vertex): this is an acceptable special situation. In all cases, a different diagonal can be added to produce another partitioning of the pseudo-quadrilateral into two pseudo-triangles. See Fig. 8. The following simple lemma shows that this is always possible.

Lemma 2.1 *A pseudo-quadrilateral can be subdivided by bitangents into two pseudo-triangles in exactly two ways.*

Proof: If we label in ccw order the four corners (convex vertices) as 1, 2, 3 and 4, there are exactly two geodesic paths inside the polygon joining opposite pairs of these corners (1 and 3, resp. 2 and 4). Each geodesic follows the boundaries of the convex chains except for one line segment, which is a bitangent. $\square$

More rigorous proofs of this and other similar simple properties needed later in the paper can be done using facts about point sets (and graphs embedded on them) that derive from the oriented matroid nature of a point set. These are the subject of the future companion paper [74].

The operation of replacing one of these two bitangents by the other is called a (diagonal) *flip* in a pseudo-quadrilateral. More generally, if any interior (non convex hull) diagonal is removed from a pointed pseudo-triangulation, it always induces a pseudo-quadrilateral and a unique flip.

2.2 Definitions: Laman Graphs and Combinatorial Rigidity

The definitions and results in this section are well known in Rigidity Theory. They are included here for a self-contained presentation of the proofs given in the next section.

Laman graphs. A graph G with n vertices and m edges is a *Laman graph* if $m = 2n - 3$ and every subset of k vertices spans at most $2k - 3$ edges. This is called the definition *by counts* and is one of the many equivalent ways in which Laman graphs can be defined.

Laman graphs are *the* fundamental objects in 2-dimensional Rigidity Theory. Also known as *isostatic* or *generically minimally rigid graphs*, they characterize combinatorially the property that a graph, embedded on a generic set of points in the plane, is infinitesimally rigid (with respect to the induced edge lengths). See Section 3.1 for rigidity theoretic definitions, and [51, 38, 82].

Henneberg constructions for Laman graphs. A Laman graph on n vertices has an inductive construction as follows (see [42, 78]). Start with an edge for $n = 2$. At each step, add a new vertex in one of the following two ways:

- **Henneberg I** (vertex addition): the new vertex is connected via two new edges to two old vertices.

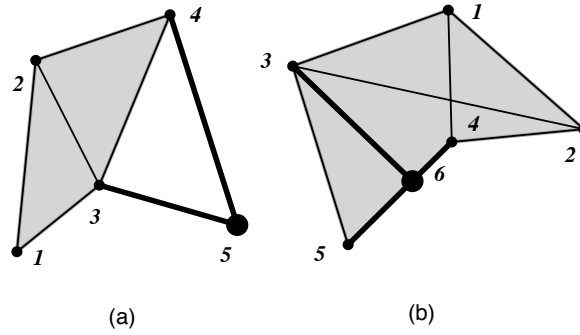


Figure 9: Illustration of the two types of steps in a Henneberg sequence, with vertices labelled in the construction order. The shaded part is the old graph, to which the black vertex is added. (a) Henneberg I for vertex 5, connected to old vertices 3 and 4. (b) Henneberg II for vertex 6, connected to old vertices 3, 4 and 5.

- **Henneberg II** (edge splitting): a new vertex is added on some edge (thus splitting the edge into two new edges) and then connected to a third vertex. Equivalently, this can be seen as removing an edge, then adding a new vertex connected to its two endpoints and to some other vertex.

See Figure 9, where we show drawings with crossing edges, to emphasize that the Henneberg constructions work for *general*, not necessarily *planar* Laman graphs.

The following result is stated by Henneberg [42], and proven by Tay and Whiteley [78].

Lemma 2.2 *A graph is Laman if and only if it has a Henneberg construction.*

In the next section, we will use a similar inductive proof technique to obtain a related inductive construction for pointed pseudo-triangulations (also called, for these historical reasons, a *Henneberg construction*).

2.3 Pointed Pseudo-Triangulations are Laman Graphs

The following theorem exhibits the combinatorial properties of pointed pseudo-triangulations which imply useful rigidity theoretic consequences: they are Laman graphs, and hence generically rigid.

Theorem 2.3 (Characterization of pointed pseudo-triangulations)

Let $G = (V, E)$ be a graph embedded on the set $\mathcal{P} = \{p_1, \dots, p_n\}$ of points. The following properties are equivalent.

1. **(Minimum Pseudo-Triangulation)** G is a minimum pseudo-triangulation of $\mathcal{P}$.
2. **(Pointed Pseudo-Triangulation)** G is a pointed pseudo-triangulation of $\mathcal{P}$.

3. (**$2n - 3$ Pseudo-Triangulation**) G is a pseudo-triangulation of $\mathcal{P}$ with $2n - 3$ edges (and, equivalently, with $n - 2$ faces).
4. (**$2n - 3$ Planar and Pointed**) The set of edges E is planar (non-crossing), pointed and has $2n - 3$ elements.
5. (**Maximal Planar and Pointed**) The edges E of G form a pointed and planar set of segments, and E is maximal (by inclusion) with this property.
6. (**Planar Pointed Henneberg-type Construction**) G can be constructed inductively as follows. Start with a triangle. At each iteration, add a new vertex in one of the faces of the already constructed embedded graph (which will be a pointed pseudo-triangulation). Connect in one of the two ways (see Fig. 10):
 - (a) Type 1: (degree 2 vertex) Join the vertex by two tangents to the already constructed part. If the new vertex is outside the convex hull, the two tangents are uniquely defined. If it is inside an internal pseudo-triangular face, there are three different ways of adding two tangents to the three inner convex chains of the face, out of which two are chosen.
 - (b) Type 2: (degree 3 vertex) Add two tangents as before. Then choose an edge on the boundary chain between the two tangent points and remove it. This creates a pseudo-4-gon. Re-pseudo-triangulate by adding the unique bitangent different from the one just removed (flip). This edge will be incident to the newly added vertex.

Moreover, if any of the above conditions is satisfied, then the subgraph induced on any subset of k vertices has at most $2k - 3$ edges (the hereditary property). Therefore G is a **Laman graph**.

For the proof, we will need some basic definitions and facts, most of which are folklore in Computational Geometry. Therefore we present them in a sketchy manner.

Facts

1. Given a convex hull and an exterior vertex, there exist exactly two tangents from the point to the hull.
2. Given a pseudo-triangle and a vertex interior to it, there exist exactly 3 tangents, all interior to the pseudo-triangle, from the point to the three inner convex chains.
3. Given a pseudo-quadrilateral, there exist exactly two ways of adding a bitangent between two inner convex chains. Each one induces a partitioning of the pseudo-quadrilateral into two pseudo-triangles. This is Lemma 2.1.
4. (*Flips in pointed pseudo-triangulations*) Any internal edge in a pointed pseudo-triangulation can be flipped: the edge is deleted and this joins two pseudo-triangles into a pseudo-quadrilateral, which can then be re-pseudo-triangulated in exactly one other way. See Lemma 2.1 and Fig. 6(c).

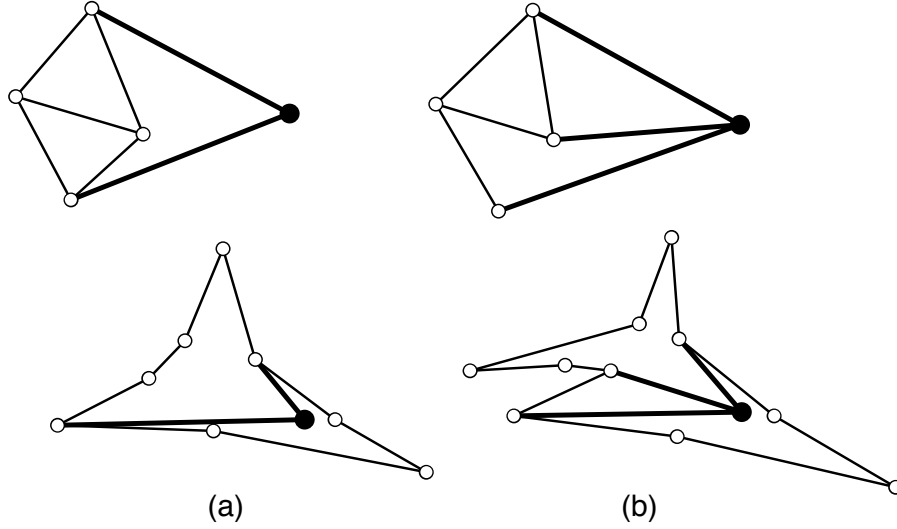


Figure 10: Henneberg steps for pointed pseudo-triangulations. (a) type 1 and (b) type 2. The newly added vertex is black. Top row, when the new vertex is added on the outside face. Bottom row, when it is added inside a pseudo-triangular face. For the type 2 step, a diagonal was previously removed.

We are now ready for the proof of Theorem 2.3.

Proof:

$1 \Leftrightarrow 2 \Leftrightarrow 3$

Let G be a pseudo-triangulation, and let $v = n$, e and f denote its number of vertices, edges and interior faces. Euler's formula gives $v - e + f = 1$. Denote by d_i the degree of vertex i and by c_i the number of convex angles (corners) incident to it. If i is a pointed vertex, $c_i = d_i - 1$, otherwise $c_i = d_i$. Let A be the set of pointed vertices, B the set of non-pointed ("bad") vertices of G and let $b = |B|$ be the total number of non-pointed vertices in G . Then

$$2e = \sum_{i=1}^n d_i = \sum_{i \in A} (c_i + 1) + \sum_{i \in B} c_i = \sum_{i=1}^n c_i + (n - b)$$

But $\sum_{i=1}^n c_i \geq 3f$ since each face has at least three corners (exactly three if it is a pseudo-triangle). Therefore $2e \geq 3f + (n - b)$. Plugging this into Euler's formula yields $e \leq 2n - 3 + b$. When each face is a pseudo-triangle, this holds with equality $e = 2n - 3 + b$ and is minimum iff $e = 2n - 3$ iff $b = 0$, i.e. when G is pointed.

$3 \Leftrightarrow 4$

One direction is obvious. For the other direction, let G be planar, pointed and with $2n - 3$ edges and hence, by Euler's formula, with $n - 2$ faces. A similar argument as above shows that the total number of corners in G is $c = \sum_{i=1}^n (d_i - 1) = 2e - n = 3(n - 2)$. Since each interior face in a planar graph has at least 3 inner convex angles, it follows that each must have *exactly* 3 convex angles, hence it is a pseudo-triangle.

$2 \Leftrightarrow 5$

If G is a pointed pseudo-triangulation, each new diagonal added to it will violate either planarity or pointedness of at least one vertex, because there are no common bitangents to the inner convex chains of a pseudo-triangle. Hence G is maximal. To prove the other direction, let G be a planar and pointed graph to which no edge can be added without violating one or the other of these two properties. Obviously G must contain all the convex hull edges, because otherwise they can be added without causing any violation of planarity or pointedness. Assume now that at least one face is not a pseudo-triangle. It is straightforward to show that each face must be topologically a disk, and hence a pseudo- k -gon for some $k \geq 3$. This is because otherwise a “face” would be a polygon with holes, to which we could always add another diagonal (a tangent, say, from a point on the outer boundary to an inner hole) while keeping the graph still pointed and planar. But if a face is a pseudo- k -gon, for some $k > 3$, then we can always add a new bitangent, contradicting maximality.

To prove that G is a *Laman graph*, we use the fact that planarity (non-crossing) and pointedness are hereditary properties: if they are satisfied on G , they are satisfied on any subset of G . We must prove that every subset of k vertices spans at most $2k - 3$ edges. Indeed, if that was not the case for some induced subgraph, then the proof of the equivalences $1 \Leftrightarrow 2 \Leftrightarrow 3$ would imply that it would violate either planarity or pointedness.

$4 (\Leftrightarrow 3) \Leftrightarrow 6$

Let G be planar, pointed and with $2n - 3$ edges. We will also use the derived fact that any subset of k vertices spans at most $2k - 3$ edges. We work out the Henneberg construction in reverse. Because the number of edges is $2n - 3$, there must exist at least one vertex of degree strictly less than 4. This cannot be 1 or 0, because then the Laman count property would be violated on the subset of the other $n - 1$ vertices. If there exists a vertex of degree 2, its two adjacent edges are tangent to the face obtained by removing them, because of pointedness. For a vertex of degree 3, the two extreme edges adjacent to it (i.e. those adjacent to the reflex angle of the pointed vertex) must be tangents to the object obtained by removing them (again, because of the property of pointedness). The face obtained by removing the third edge (before removing the two extreme ones) is a pseudo-quadrilateral (this follows from the other equivalences), and the addition of the second bitangent recreates a pseudo-triangle. Removing the vertex, the remaining graph satisfies the same properties (because of the *hereditary property*). Hence the argument continues.

The other direction is straightforward: at each step in a Henneberg-type construction, the number of vertices increases by 1, the number of edges by 2 and the graph remains planar and pointed. $\square$

In the rest of this paper we will only be concerned with *pointed* pseudo-triangulations, even if we may omit the word *pointed* to simplify the terminology.

Corollary 2.4 *The underlying graphs of pointed pseudo-triangulations are generically minimally rigid graphs.*

Not all generically minimally rigid graphs have embeddings as pseudo-triangulations, because not all are planar graphs. The smallest example is $K_{3,3}$. However, a recent result [41] shows that all minimally rigid planar graphs can be embedded as pseudo-triangulations.

Laman graphs have further nice combinatorial properties, which are inherited by pseudo-triangulations. For completeness, we give them here (but will not make any further use of them.)

Theorem 2.5 (Tree decompositions of pseudo-triangulations)

1. **Two spanning tree decomposition.** *If any edge is doubled, a pseudo-triangulation can be decomposed into two spanning trees.*
2. **2T3 decomposition.** *A pseudo-triangulation can be decomposed into three disjoint trees, so that each vertex is adjacent to exactly two of them.*

Proof: Part 1 is a direct consequence of the result of Lovasz and Yemini [53] for generically minimally rigid graphs (Laman graphs). Part 2 is a consequence of Crapo’s theorem [32]. $\square$

3 Rigidity of Pseudo-Triangulations

We have shown that (pointed) pseudo-triangulations are Laman graphs, which are rigid in almost all embeddings (“generically”, see [36]). We prove now that pseudo-triangulations have an even more special property: they are *always* infinitesimally rigid.

We start by defining the needed concepts from Rigidity Theory. Although these results are not new, the presentation is. We emphasize the role of the number of edges on the infinitesimal rigidity and self-stress properties of a graph embedding. This perspective, based on rank and orthogonality relations from elementary linear algebra, is then used in the proof of infinitesimal rigidity for pointed pseudo-triangulations.

We have chosen the sequence of facts in such a way that it is clear what properties hold for general frameworks sharing some combinatorial properties with pseudo-triangulations (such as edge counts or planarity). This way, it is easy to extract what is specific to pointed pseudo-triangulations, and generalize to other situations, when needed. Indeed, in the companion paper [74] we generalize some of these properties to the oriented matroid setting.

3.1 Frameworks and Rigidity

Convention. From now on, we will use interchangeably $\mathcal{P} = (p_1, \dots, p_n) \in (\mathcal{R}^2)^n$, with $p_i = (x_i, y_i)$ or its *flattened* version $p = (x_1, y_1, \dots, x_n, y_n) \in \mathcal{R}^{2n}$ to stand for elements of either $(\mathcal{R}^2)^n$ or $\mathcal{R}^{2n}$.

Frameworks and Configuration Spaces. A *bar-and-joint framework* (or a *fixed edge-length framework*, or shortly, a *framework*) (G, L) is a graph $G = (V, E)$, $|V| = n$, together with a set of strictly positive weights $L = \{l_e | e \in E\}$, $(l_e \in \mathcal{R}, l_e > 0)$ meant to be used as *edge lengths*. A *realization* $G(\mathcal{P})$ of (G, L) on a set of points $\mathcal{P} = \{p_1, \dots, p_n\} \subset \mathcal{R}^2$ is a mapping $i \mapsto p_i$ of vertices to points and edges to line segments (i.e. an embedding of G in the plane) so that the length $\|p_i - p_j\|$ of the segment $\overline{p_i p_j}$ corresponding to the edge $e = ij$ is equal to l_e . Since each realization comes together with a whole set of other realizations obtained from it by translations and rotations, it is convenient to factor out the rigid motions of the whole plane. This can be done, for instance, by *pinning down* an edge, e.g. if $e = 12 \in E$ we will

set $x_1 = y_1 = 0$, $x_2 = l_{12}$, $y_2 = 0$. The set of all possible realizations of a framework, with rigid motions factored out (e.g. via an arbitrary pinned down edge) is called its *configuration space*. Its topological and differential properties do not depend on the choice of the pinned down element.

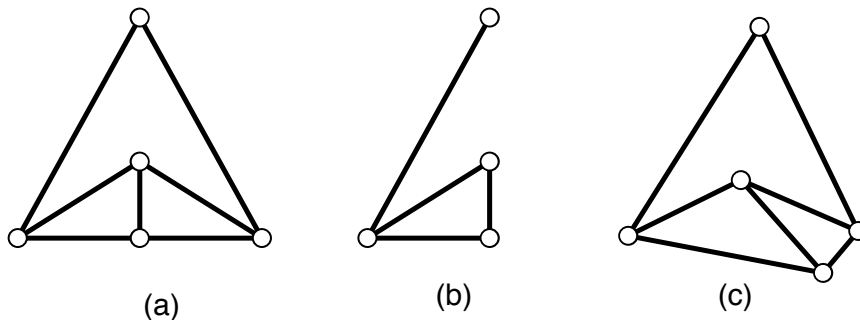


Figure 11: A framework with two distinct embeddings, one of which (a) is rigid and the other (b) is flexible, with one degree-of-freedom. Notice that the symmetry of (a) allows in (b) for some pairs of vertices to be mapped to identical points, and some pairs of edges to be drawn one on top of the other. This allows in (b) for two identical edges, embedded on top of each other, to move with one degree of freedom, as a single dangling edge does. (c) The same graph for other edge lengths has only rigid embeddings.

In this paper we are *not* concerned with questions of realizability, as we will always start with a given embedding, from which the edge lengths are actually computed if needed. Therefore the actual values of the edge lengths are not relevant to our discussion: the configuration space will always be non-empty. To simplify the terminology, from now on we will usually refer to a realization $G(\mathcal{P})$ as a *framework*, and when we actually mean (G, L) , we will say it explicitly.

Rigid and flexible frameworks. The configuration space of a (fixed edge-length) framework (G, L) may be disconnected and in general may have a complicated topological structure. The *dimension* of the component of the configuration space in which a framework (realization) $G(\mathcal{P})$ resides is called its *number of degrees of freedom*. If that component is a single, isolated point, the framework $G(\mathcal{P})$ is called *rigid*, otherwise it is *flexible*. We emphasize that rigidity and flexibility refer only to the configuration space component to which *a certain realization* $G(\mathcal{P})$ belongs. Indeed, there exist frameworks (G, L) for which different components may have different dimensions, see Fig. 11.

Minimal Rigidity. A rigid framework is *minimal* if the removal of any edge makes it flexible. Otherwise we say that it is *overbraced*. The example in Fig. 11(c) is minimally rigid. Any extra edge overbraces it.

Generically, the minimum number of bars needed to induce a rigid framework is $m = 2n - 3$. Indeed, the configuration space is given by m quadratic equations in $2n$ variables $\{x_1, y_1, \dots, x_n, y_n\}$, so $2n$ independent equations are needed to get a discrete set of solutions. We need three equations to eliminate the rigid motions, and the other $2n - 3$ must be independent bar-lengths equations of the form $(x_i - x_j)^2 + (y_i - y_j)^2 = l_{ij}^2$. Therefore generically we expect that

$2n - 3$ independent edges and edge-length values will typically produce a zero-dimensional configuration space; more than $2n - 3$ edges will typically produce an empty configuration space, except for very special, dependent values for some of the edge lengths; and less than $2n - 3$ edges will induce flexible frameworks, with a number of degrees of freedom equal to $2n - 3 - m$. The main difficulty in analyzing rigidity properties of frameworks is due to the occurrence of non-generic situations.

Our main rigidity theoretic results on pseudo-triangulations and pseudo-triangulation mechanisms (sections 3.2 and 4.5) may be interpreted as saying that planarity and pointedness (as in *pointed pseudo-triangulations* and subsets of their edge set) induce *generic frameworks*: their configuration spaces have the generic dimension for the corresponding number of edges.

Infinitesimal Rigidity. An equivalent way of saying that a framework $G(\mathcal{P})$ is *flexible* is that its vertices can be moved *continuously* while preserving the lengths of the edges, such that the motion is not a trivial rigid motion (translation and rotation.) The non-trivial motion is called a *flex* or *reconfiguration* of the framework. It is a continuous curve (one dimensional trajectory) $p(t) = \{p_1(t), \dots, p_n(t)\}$ in the configuration space going through the point $\mathcal{P}$ giving the framework realization, such that at each moment in time:

$$\|p_i(t) - p_j(t)\| = l_{ij} \quad (1)$$

for all edges $ij \in E$. If we assume that the flex $p(t)$ has other good analytic properties (e.g. is differentiable), then taking the derivative of equation (1) we obtain the conditions:

$$\langle p_i(t) - p_j(t), p'_i(t) - p'_j(t) \rangle = 0, \forall ij \in E \quad (2)$$

Here $\langle, \rangle$ is the dot product. The first-order derivative $p'(t)$ of the motion $p(t)$, computed at a given moment in time t , is called an *infinitesimal* or *instantaneous motion at time t* . These considerations motivate the following definition.

For a given framework $G(p)$, an *infinitesimal motion* $v = \{v_1, \dots, v_n\}$ is an assignment of a velocity v_i to each point p_i such that the lengths of the edges are preserved:

$$\langle p_i(t) - p_j(t), v_i(t) - v_j(t) \rangle = 0, \forall ij \in E \quad (3)$$

A framework is *infinitesimally* (or *first-order*) *flexible* if it has an infinitesimal motion, otherwise it is *infinitesimally* (or *first-order*) *rigid*.

Infinitesimally rigid frameworks are rigid, but the opposite statement may not always be true. See Fig. 12 (a). To capture the relationship between rigidity and infinitesimal rigidity, flexibility and infinitesimal flexibility of $G(\mathcal{P})$, one must investigate the *differential* properties of its configuration space in a neighborhood of the particular point (realization) $G(\mathcal{P})$.

The Edge Map. Given a graph G with n vertices and $|E| = m$ edges, the *edge map* or *rigidity map* $f_G : \mathcal{R}^{2n} \mapsto \mathcal{R}^m$ associates to a point set $p = (p_1, \dots, p_n) \subset \mathcal{R}^{2n}$ the vector of the squared lengths of the edges $ij \in E$, in some fixed, predefined order:

$$f_G(p_1, \dots, p_n) = (\|p_i - p_j\|^2)_{ij \in E} \quad (4)$$

We say that a point $p \in \mathcal{R}^{2n}$ is a *regular* or *generic point* of f_G if the rank of the differential of f_G is maximum at p :

$$\text{rank } df_G(p) = \max\{\text{rank } df_G(q) : q \in \mathcal{R}^{2n}\} \quad (5)$$

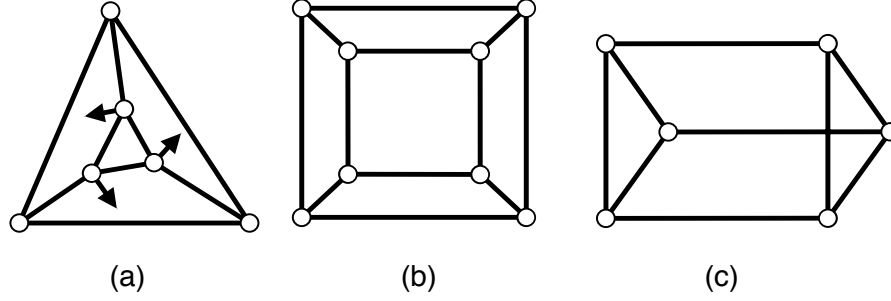


Figure 12: (a) A rigid framework which is not infinitesimally rigid. (b) An infinitesimally flexible framework which is rigid. (c) A generically rigid graph in a flexible embedding.

Otherwise, p is a *singular point*.

If $m \geq 2n - 3$, we expect the rank to be $2n - 3$ for regular (generic) points. For $m < 2n - 3$, the generic rank is m . The rank drops at singular points. The following closer analysis relates it to the infinitesimal rigidity properties of the underlying framework $G(\mathcal{P})$.

The Rigidity Matrix. The rigidity matrix $M_G(\mathcal{P})$ (shortly M) associated to an embedded framework $G(\mathcal{P})$ is the Jacobian matrix $df_G(p)$ of the edge map at the point $p \in \mathcal{R}^{2n}$ corresponding to the embedding $\mathcal{P} = (p_1, \dots, p_n)$. It has a row for each edge $ij \in E$ and 2 columns for each vertex. The row indexed by the edge $ij \in E$ has 0 entries everywhere, except in the i th and j th group of 2 columns, where the entries are $p_i - p_j$, resp. $p_j - p_i$.

$$ij \begin{pmatrix} 1 & \cdots & i & \cdots & j & \cdots & n \\ 0 & \cdots & p_i - p_j & \cdots & p_j - p_i & \cdots & 0 \end{pmatrix}$$

The linear subspace of *infinitesimal motions* $v \in \mathcal{R}^{2n}$ of the framework is the kernel (null space) of M , $\ker M = \{v \in \mathcal{R}^{2n} \mid Mv = 0\} \subset \mathcal{R}^{2n}$. A vector of velocities $(v_1, \dots, v_n)$ will be interchangeably written as a flattened vector $v \in \mathcal{R}^{2n}$. Any such v (not necessarily an infinitesimal motion) yields a set of values associated with the edges of E , $d_{ij} = \langle p_i - p_j, v_i - v_j \rangle$, $\forall ij \in E$. This set is a linear subspace, the image space of M , $\text{Im} M = \{d \mid d = Mv\} \subset \mathcal{R}^{|E|}$. The sign of d_{ij} has a physical interpretation: if positive, v is an infinitesimal motion which *expands* the edge, i.e. increases the distance between the vertices i and j . The other two important linear subspaces associated with M are the null space $\ker M^T$ of the transposed matrix M^T and its image space $\text{Im} M^T$. They also have physical interpretations, as follows.

Self stresses. A *self-stress* on a framework $G(\mathcal{P})$ is an assignment of scalars w_{ij} to edges $ij \in E$ such that the *forces* along the edges around each vertex, scaled by the self-stresses, are *in equilibrium*: $\forall i \in V$, $\sum_{ij \in E} w_{ij}(p_i - p_j) = 0$. Self stresses form a linear subspace $\ker M^T = \{w = (w_{ij})_{ij \in E} \mid M^T w = 0\} \subset \mathcal{R}^{|E|}$, which is orthogonal to $\text{Im} M$:

$$\ker M^T \perp \operatorname{Im} M \quad (6)$$

A *non-trivial self-stress* is one with at least one non-zero component $w_{ij} \neq 0$ on some edge $ij \in E$.

A vector of scalar values associated to the edges $w = (w_{ij})_{ij \in E}$, not necessarily a self-stress, induces a vector of velocities associated with the vertices, $\{v \mid v = M^T w\} \subset \mathcal{R}^{2n}$. Each v is componentwise a sum of the incident edge vectors, scaled by the w_{ij} factors: $v_i = \sum_{ij \in E} w_{ij}(p_i - p_j)$. They form a linear subspace $\operatorname{Im} M^T = \{v \mid v = M^T w, w \in \mathcal{R}^{|E|}\} \subset \mathcal{R}^{2n}$, which is orthogonal on the subspace of infinitesimal motions $\ker M$:

$$\ker M \perp \operatorname{Im} M^T \quad (7)$$

The rank relations. The following well-known relations hold between the dimensions of the four linear spaces associated with the $2n \times m$ matrix M :

$$\begin{aligned} \dim \ker M + \dim \operatorname{Im} M^T &= 2n \\ \dim \ker M^T + \dim \operatorname{Im} M &= m \\ \dim \operatorname{Im} M &= \dim \operatorname{Im} M^T \end{aligned} \quad (8)$$

Minimal Infinitesimal Rigidity. The kernel of the rigidity matrix $\ker M$ always contains the 3-dimensional linear subspace of trivial infinitesimal motions, translations and rotations.

The following Lemma is a simple consequence of the rank relations, and summarizes the main consequence of the lack of self-stress for graphs with $2n - 3$ edges. This is the minimum number of edges that allows for infinitesimal rigidity.

Lemma 3.1 *If $m = 2n - 3$, $\dim \ker M = 3$ iff $\dim \ker M^T = 0$. In this case, the framework $G(\mathcal{P})$ is infinitesimally rigid iff it has no self-stress.*

Proof: Using the rank relations (8), we infer that $\dim \ker M = 3$ iff $\dim \operatorname{Im} M^T = 2n - 3$, but $\operatorname{Im} M^T = \operatorname{Im} \operatorname{Im} M$, and further equal to $m - \dim \ker M^T$. Since $m = 2n - 3$, this means $\dim \ker M^T = 0$. $\square$

When the number of edges drops below $2n - 3$, non-trivial infinitesimal motions are unavoidable.

Lemma 3.2 *If $m \leq 2n - 4$ then $G(\mathcal{P})$ always has a non-trivial infinitesimal motion.*

When the number of edges exceeds $2n - 3$, self-stress is unavoidable.

Lemma 3.3 *If $m \geq 2n - 2$ then $G(\mathcal{P})$ always has a non-trivial self-stress.*

Generic Minimal Rigidity. We focus now on graphs with exactly $2n - 3$ edges. If such a graph has an infinitesimally rigid embedding $G(\mathcal{P})$, then G is not only rigid at $\mathcal{P}$, but also on any point set $\mathcal{P}'$ belonging to a small neighborhood of $\mathcal{P}$. Because of the edge count, this is equivalent

to $G(\mathcal{P}')$ being self-stress-free on the same neighborhood. Moreover, G satisfies the property *minimally*: the removal of any edge turns it into a graph which is no longer infinitesimally rigid.

We say that $G(\mathcal{P})$ is a *generic rigid embedding* of a graph G with $2n - 3$ edges if it is infinitesimally rigid (or, equivalently, self-stress-free) on an open neighborhood of $\mathcal{P}$. Such a point $\mathcal{P}$ is a regular point of the edge map for G .

A graph G with $2n - 3$ edges is called a *generically (minimally) rigid graph* if it has a generic infinitesimally rigid embedding.

Theorem 3.4 (Laman’s Theorem [51]) *G is generically minimally rigid iff G is a Laman graph.*

Self-stresses in Planar Frameworks and Maxwell’s Theorem. The following beautiful connection between the existence of non-trivial self-stresses and 3d-liftings of *planar* graphs goes back to Maxwell [55, 56]. Some topological details, as well as the extension to signed stresses, have been fixed later (see [33]). We present here only the simplified formulation that will be used later in our proofs.

A 3d-lifting of a planar (non-crossing) framework $G(\mathcal{P})$ is an assignment of a *height* $z_i \in \mathcal{R}$ to each vertex i such that when the graph G is *lifted* in 3d on the points $(p_i, z_i) := (x_i, y_i, z_i)$, the vertices of each face of G are coplanar. We may assume without loss of generality that the outer face of G stays in the original plane ($z_i = 0$ for vertex p_i on the outer face), and the other vertices are lifted. A lifting is trivial if all z_i ’s are 0. We are interested only in non-trivial liftings.

Notice that such a lifting (in the particular case of a non-crossing framework embedding presented here) is a polyhedral terrain. The outer (unbounded) face is assumed to be flat at level $z = 0$.

Two faces adjacent along an edge ij span a dihedral angle which, when viewed from below, may be convex, reflex or equal to π . Call the edge a *mountain* if this angle is convex, *valley* if reflex and *flat* if equal to π .

Not all planar graph frameworks admit non-trivial 3d-liftings. Maxwell’s theorem characterizes those which do.

Theorem 3.5 (Maxwell [55, 56], Crapo and Whiteley [33]) *A planar (non-crossing) framework has a non-trivial 3d lifting if and only if it has a non-trivial self-stress. Moreover, the correspondence between self-stresses and liftings maps the edges with positive self-stress to mountain edges, those with negative stress to valley edges and those with zero stress to flat edges.*

Bow’s construction and Maxwell liftings for non-planar graphs. Maxwell’s theorem can be extended to non-planar frameworks by applying a simple trick, called *Bow’s construction*, see [57]. Let $G(\mathcal{P})$ be a framework where some of the edges have proper crossings. We will introduce a new vertex for each crossing of two edges. Then we “split” each of the two crossing edges into two new edges, each one having the new vertex as an endpoint. It can be easily proven that the resulting framework inherits the self-stress properties of the original framework, including the signs of the split edges. See [57, 33], and Figure 13(b) and (c).

3.2 Pointed Pseudo-Triangulations are Infinitesimally Rigid

We prove now the main rigidity theoretic property of pointed pseudo-triangulations: they are always infinitesimally rigid. The main tool used in the proof is Maxwell’s theorem. The same

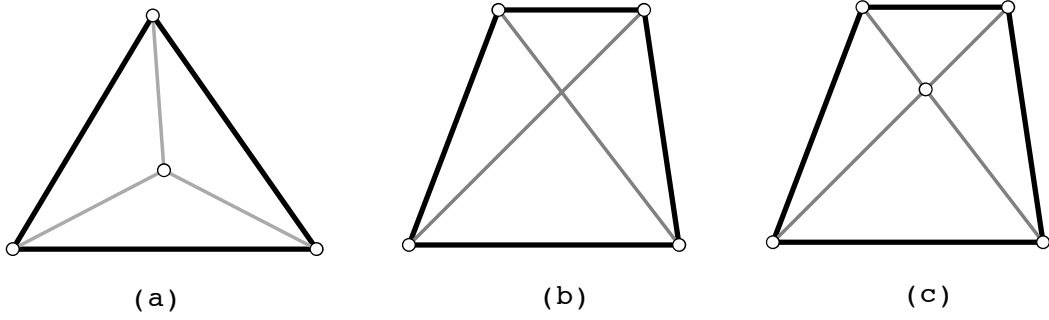


Figure 13: (a) The smallest example of a planar self-stressed graph with no collinear vertices. It lifts to a tetrahedron. The thick, convex hull edges support a positive self-stress, and are valley edges in the lifting. The internal, thin grey edges, support a negative self-stress and lift to mountain edges. (b) The same graph, in a non-planar embedding, and the corresponding self-stress. (c) Bow's construction applied to the embedding in (b): a new vertex is added at the cross-point of the two internal diagonals. The signs of the self-stresses is inherited from the original graph. The self-stressed *planar* graph in (c) lifts to a pyramid.

tool will be used in the next section to prove that pseudo-triangulations with a convex hull edge removed are expansive mechanisms.

Theorem 3.6 *Pointed pseudo-triangulations are always infinitesimally rigid.*

Let us reformulate the theorem, to emphasize all the properties of pointed pseudo-triangulations that the proof will exhibit.

Theorem 3.7 (Infinitesimal Rigidity of Pointed Pseudo-Triangulations) *A pointed pseudo-triangulation $G(\mathcal{P})$ is a planar Laman graph embedding which is always infinitesimally rigid and self-stress-free.*

In the proof, we will make use of the following simple fact.

Lemma 3.8 *For any lifting into 3d at non-constant height of a flat polygon, all vertices of maximum height are corners.*

Proof: All vertices on the convex hull of a polygon are corners. The maximum height is attained on the convex hull of a 3d object, which, for the flat lifting considered here, is exactly the 2d convex hull. See Fig. 14. $\square$

Proof of Theorem 3.7

Because G is a Laman graph, and hence has $2n - 3$ edges, we use Lemma 3.1 to prove infinitesimal rigidity by showing that $G(\mathcal{P})$ is self-stress-free.

For the sake of a contradiction assume that there exists a non-trivial self stress $(w_e)_{e \in E}$, $w_e \neq 0$ for some $e \in E$. Then, by Maxwell's Theorem, there exists a non-trivial 3d-lifting which

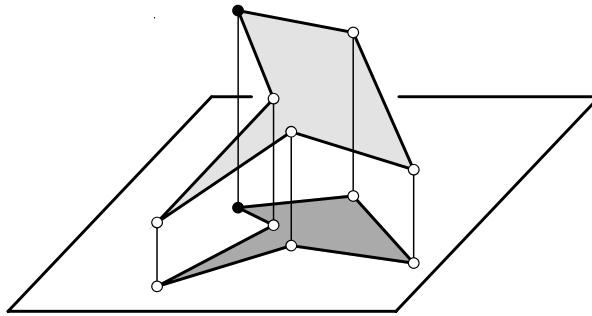


Figure 14: Illustration of Lemma 3.8: the vertex of maximum z -coordinate of the lifted face projects to a vertex on the convex hull of the flat polygon. Reflex vertices cannot lift to maxima.

is not flat on the edges with non-zero self-stress. Consider the points of local maximum height in the lifting. If the set of these points contains only vertices of the convex hull of $G(\mathcal{P})$, it means that the whole lifting lies underneath the $z = 0$ plane. We can simply reverse all the signs to make it above the plane, and assume that we have a local maximum which is not a convex hull vertex and lies above the xy -plane. This is strictly above the xy -plane, since the self-stress is not trivial. We will derive a contradiction by considering one of these local maxima.

Consider the convex hull of the maximum region and a vertex on this convex hull. By Lemma 3.8, this vertex must be a corner of all the faces incident to it, hence its projection is a corner in all incident faces. This contradicts the pointedness of the pseudo-triangulation.

Since the lifting is impossible, the graph is self-stress-free and hence infinitesimally rigid. $\square$

Corollary 3.9 *A pointed pseudo-triangulation $G(\mathcal{P})$ is always infinitesimally rigid, and minimally so: if any edge is removed, it becomes infinitesimally flexible.*

4 Expansive 1dof mechanisms from pseudo-triangulations

In this section we turn our attention to flexible frameworks, specifically those obtained by removing an edge from a Laman graph in a generic embedding. We show that we always obtain a 1dof mechanism: a framework with a one-dimensional configuration space. A pointed pseudo-triangulation with a convex hull edge removed will be called a *pseudo-triangulation mechanism* or shortly a *ppt-mechanism*.

The main theorem of this section is that ppt-mechanisms are *expansive* and smooth for a bounded open interval of their one-dimensional configuration space. This interval is characterized combinatorially by all the realizations of the underlying framework having the same combinatorial structure, more precisely, what we call the *combinatorial pseudo-triangulation* (defined in this section). These pseudo-triangulation mechanisms will be used in Section 6 as the building blocks of the expansive convexifying trajectory of our solution to the Carpenter's Rule problem.

4.1 Preliminaries: Laman graphs with a removed edge

We first investigate the combinatorial structure of graphs obtained by removing an edge from a Laman graph. We show that they have a natural decomposition into disjoint Laman subgraphs, called *rigid components*.

Laman-minus-one graph. This is a graph obtained by removing an edge from a Laman graph. It has $2n - 4$ edges, and each subset of k vertices spans at most $2k - 3$ edges.

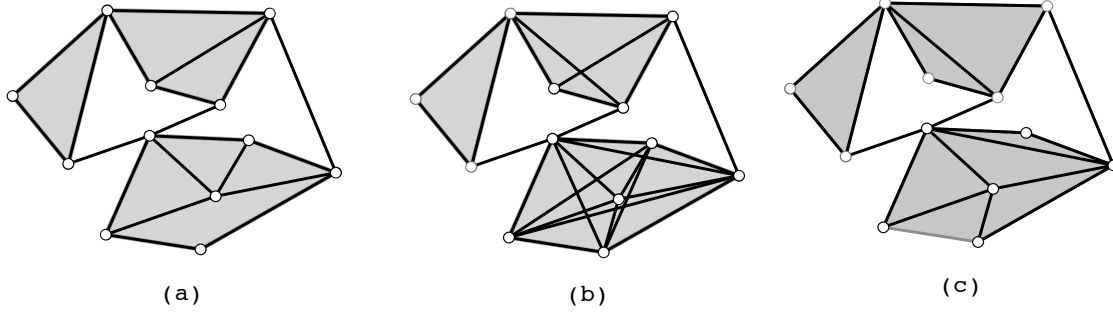


Figure 15: (a) The decomposition of a Laman-minus-one graph into rigid components, six in this case: the three big grey blocks, and the three remaining edges. (b) The collapsed Laman-minus-one graph induced by (a). (c) An equivalent Laman-minus-one graph, differing from (a) on the rigid components (the graph edges are black and thick).

Rigid components in Laman-minus-one graphs. In a Laman-minus-one graph, consider maximal subgraphs $G_R = (V_R, E_R)$ spanning exactly $|E_R| = 2|V_R| - 3$ edges. Since subsets of V_R satisfy the Laman count property, such a subgraph G_R is a Laman graph, called a *rigid component* (*r-component*) of G .

Lemma 4.1 *The edge set of any Laman-minus-one graph is partitioned into (disjoint) rigid components.*

Proof: A rigid component is well defined, by maximality. Two rigid components can share a vertex, but not more. Indeed, if there is no edge between the two common vertices, the union of the two r-components would violate the Laman count. Otherwise, if there is an edge, maximality would be violated, as in this case the union of the two r-components would be a larger Laman subgraph. $\square$

See Figure 15(a) for an example.

Collapsed Laman-minus-one graphs. Intuitively, in each rigid component all the diagonals, present or missing, are rigid. Hence, although a Laman-minus-one graph will (generically) be flexible, its r-components will move rigidly. If we replace each rigid component in a Laman-minus-one graph by the complete graph, the resulting graph is called a *collapsed Laman-minus-one graph*. See Figure 15(b) for an example.

Equivalent Laman-minus-one graphs. Two Laman-minus-one graphs are called *equivalent* if they differ only in the choice of Laman subgraphs spanning certain rigid components. In other words, the collapsed graphs of two equivalent Laman-minus-one graphs are the same. See Figure 15 (a,c).

4.2 1dof Mechanisms: infinitesimal flexibility and smooth trajectories

We now investigate the rigidity-theoretic properties of Laman-minus-one graphs. Generically, they are flexible, with one-degree-of-freedom. We identify the properties of their smooth intervals, and the unpleasant role of singularities in their configuration spaces. In the next section 4.3 we will use these considerations to justify the investigation of smoothness properties for pseudo-triangulation mechanisms and their role in the convexification algorithm.

Minimal Infinitesimal 1dof Flexibility. Consider a Laman-minus-one graph: it has $2n - 4$ edges. The kernel $\ker M$ of the rigidity matrix M always contains the 3-dimensional linear subspace of trivial infinitesimal motions, translations and rotations.

The proof of the following Lemma follows along the same lines as the proof of Lemma 3.1, and is a simple consequence of the rank relations (8). It shows the basic consequences of the lack of self-stress for such graphs.

Lemma 4.2 *If $m = 2n - 4$, $\dim \ker M = 4$ iff $\dim \ker M^T = 0$. In this case, the framework $G(\mathcal{P})$ is infinitesimally flexible with one-degree-of-freedom iff it has no self-stress.*

In particular, for the Lemma 4.2 to be satisfied, G must be a Laman-minus-one graph and each of its r-components must be in generic (self-stress-free) embeddings (as Laman (sub)graphs). But this condition is not sufficient. All the rigid components of the Laman-minus-one graph in Figure 12 (b) are edges, which are always generic (if embedded on two distinct endpoints), but the graph itself is not in a generic embedding, since it has a self-stress.

One-degree-of-freedom mechanisms. The above considerations proved that Laman graphs with a missing edge, in generic embeddings, have a one-dimensional space of infinitesimal motions. Moreover, nearby embeddings are also generic. In fact, the following stronger property holds.

Theorem 4.3 (Main Theorem of 1dof mechanisms) *Let G be a Laman-minus-one graph, and assume that it has a generic (regular, stress-free) embedding $G(\mathcal{P})$. Then the component of its configuration space containing the given generic embedding is one-dimensional, and smooth in the neighborhood of the given generic embedding.*

Proof: The statement is a direct consequence of Proposition 2, page 282 in [12], which relies on the Inverse Function Theorem (see also [13]). Alternatively, it follows from the fundamental theorem of Ordinary Differential Equations, which in turn is a direct consequence of the Implicit Function Theorem. See Chapter 2 in [10], Corollaries 1 (existence) and 2 (uniqueness) on page 93 to the Fundamental Theorem 1 on page 89. $\square$

A framework satisfying the conditions of the previous theorem is called a *1dof mechanism*. A simple example is the Laman graph $K_{3,3}$ with a missing edge from Fig.16(a). In the special embedding of Fig.16 (b), it is the classical Peaucellier linkage, used to convert between circular

and linear motion. This embedding is special, in the sense that the edge lengths are chosen to have very specific relationships between them, while in Fig.16(a) they are as general as possible. The previous theorem does not apply to the framework in Fig. 12(b), which is *not* flexible: the component of its configuration space to which it belongs contains only this realization, which is not generic.

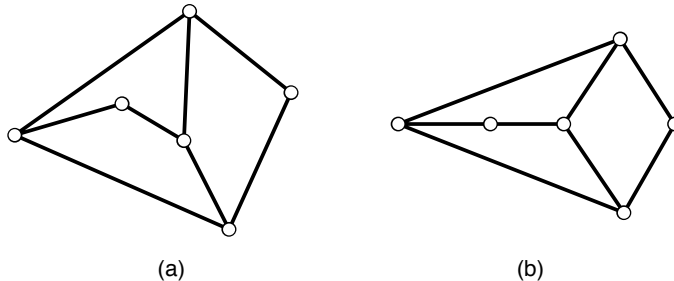


Figure 16: The classical Peaucellier linkage (b), and a general, generic embedding (a) of the underlying graph.

Given a 1dof mechanism $G(\mathcal{P})$, consider the component of its configuration space to which it belongs. We will refer to this component as the *lifetime* or *lifespan* of the mechanism $G(\mathcal{P})$, and to any configuration in the lifetime of a mechanism as a *snapshot*.

Topology of the configuration space of a 1dof mechanism. The topology of the configuration space component of a 1dof mechanism is also relevant. If there are no singular points, it is topologically a circle. But in general it may contain singularities. The simplest example is the four-bar mechanism of Fig.17. Its configuration space is smooth, unless there is a position of alignment for the four joints. This is a consequence of a general theorem of [45] regarding singularities in configuration spaces of planar polygons.

In this case, the one-dimensional configuration space component is topologically a figure-eight instead of being a smooth circle, so at the singular point (the crossing of the two branches of the figure-eight, corresponding to the alignment of the joints of the mechanism), the trajectory is not well determined. The four-bar mechanism from Fig.17 is shown in several embeddings as a pointed pseudo-triangulation with a convex hull edge removed. Its configuration space is smooth everywhere, except at the alignment position, where it is a degenerate pseudo-triangle with all corner angles equal to zero.

Such singularities in the configuration spaces of 1dof mechanisms are troublesome: how do we decide which branch to follow, and how do we enforce algorithmically such a decision? If there are no singularities, however, there are only two directions in which such a mechanism can move. In the next section 4.3 we will prove that pseudo-triangulation mechanisms are smooth as long as the combinatorial structure of the embedding is maintained. This is exactly what the algorithm in section 6 guarantees. Therefore such problems related to singularities will not occur for the mechanisms of interest in this paper.

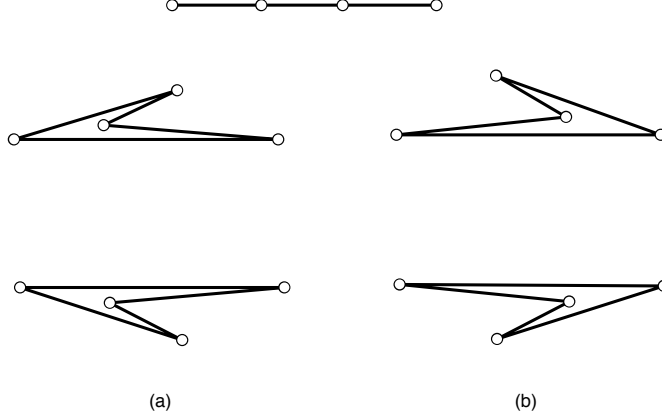


Figure 17: A semi-simple degenerate (collinear) 4-gon and the four possible ways in which a motion could go from the singular aligned position.

4.3 Expansion properties of 1dof Mechanisms

We start looking at the *free diagonals* of a Laman-minus-one mechanism: those which do not maintain rigidly their length, during a motion of the mechanism. They may expand or contract, i.e. increase or decrease in length. In this section, we investigate the expansion pattern of Laman-minus-one mechanisms.

The expansion pattern of a 1dof mechanism. Let $G(\mathcal{P})$ be a generic embedding of a Laman graph without an edge e . The space of infinitesimal motions is one-dimensional. Consider an infinitesimal motion v which *expands* the edge e , i.e. $d_e := \langle p_i - p_j, v_i - v_j \rangle > 0$. Such a motion always exists: take any motion v and if it doesn't expand, replace it with $-v$. The collection $s = (s_{ij})_{ij \in E}$ of the expansion signs of all the edges $ij \in E$, $s_{ij} := \text{sign} \langle p_i - p_j, v_i - v_j \rangle$, is called the *expansion pattern* or the *expansion signature* of $G(\mathcal{P})$. The sign is zero on edges of G and on edges *implied* by G , i.e. edges between vertices belonging to the same rigid component of G , and non-zero elsewhere.

The following is a straightforward consequence of continuity and of Theorem 4.3.

Corollary 4.4 *The expansion pattern of a generic 1dof mechanism $G(\mathcal{P})$ is the same for all the realizations in a sufficiently small neighborhood of $\mathcal{P}$.*

Proof: The expansion pattern depends continuously on the points and their velocities, which in turn depend on the rigidity matrix. All these values change continuously in a sufficiently small neighborhood of $\mathcal{P}$, where the rigidity matrix has maximal rank. Hence the signs do not change in a small neighborhood. $\square$

The following simple Lemma shows that the expansion pattern is the same for equivalent Laman-minus-one graphs embedded on the same set of points (as it is for the whole collapsed Laman-minus-one mechanism).

Lemma 4.5 *Let $G(\mathcal{P})$ and $G'(\mathcal{P})$ be generic realizations of equivalent Laman-minus-one graphs on the same point set $\mathcal{P}$. Then $G(\mathcal{P})$ and $G'(\mathcal{P})$ have the same expansion pattern.*

Proof: We replace an infinitesimally rigid component by another one with the same property: the expansion sign is zero on all the edges between vertices in the same rigid component. The space of infinitesimal motions remains the same, and hence so does the expansion sign on all the other edges. $\square$

Theorem 4.6 (Smooth interval of an expansive pattern)

A bar-and-joint framework $G(\mathcal{P})$ based on a Laman-minus-one graph G in a generic embedding is a flexible 1dof mechanism maintaining its expansion pattern in an open interval containing $\mathcal{P}$. The interval is bounded by two special embeddings of G , each one corresponding to either some free edge(s) acquiring a zero-expansion value, or to a singularity (in the component of the configuration space of G containing $G(\mathcal{P})$).

Proof: The proof of Corollary 4.4 shows that the maximal interval consisting of self-stress-free realizations of G with the same expansion pattern as $G(\mathcal{P})$, and containing $G(\mathcal{P})$, is well defined. Its endpoints consist either in points where $G(\mathcal{P})$ acquires a self-stress (and hence the rank of the rigidity matrix drops, and the configuration space becomes singular), or where the expansion sign on some free edge becomes zero. $\square$

Expansive 1dof mechanisms. A 1dof infinitesimal mechanism is (infinitesimally) *expansive* if there exists an infinitesimal motion such that all the free edges increase infinitesimally. I.e., there exists an assignment of velocity vectors v_i to each vertex p_i so that $\langle p_i - p_j, v_i - v_j \rangle \geq 0$, $\forall i, j$: the expansion pattern consists only in 0's and +'s.

Notice that the expansive property is defined *infinitesimally*. A 1dof mechanism $G(\mathcal{P})$ cannot be expansive throughout its lifetime, but only for a certain interval of time. We are interested in *generically infinitesimally expansive* 1dof mechanisms. Indeed, if $G(\mathcal{P})$ is expansive at $\mathcal{P}$, and this is a generic framework, then it will be expansive in a neighborhood of $\mathcal{P}$. Theorem 4.6 implies in this case that the configuration space is smooth and expansive in a neighborhood of $\mathcal{P}$. An *expansive mechanism* is an infinitesimally expansive 1dof mechanism $G(\mathcal{P})$, together with the maximal interval I around $\mathcal{P}$ in which it is infinitesimally expansive. We summarize these considerations in the following theorem, which will be used in section 6..

Theorem 4.7 *Let G be a Laman graph with a missing edge and let $G(\mathcal{P})$ be a generic infinitesimally flexible realization. If $G(\mathcal{P})$ is infinitesimally expansive, then G is an expansive mechanism in an open interval around $G(\mathcal{P})$, bounded by either a singularity or by an embedding where a free edge acquires zero-expansion.*

For example, the Peaucellier linkages in Fig.16 are not expansive, in the given realizations. Fig. 18 illustrates several snapshots in the lifetime of a 1dof mechanism. The first one is a semi-simple pointed pseudo-triangulation, and will be analyzed later. The second is a pointed pseudo-triangulation and is expansive. The last one is not infinitesimally expansive. The first and the third, Fig. 18 (a,c), lie at the boundary of the interval in which the mechanism in Fig. 18 (b) is infinitesimally expansive.

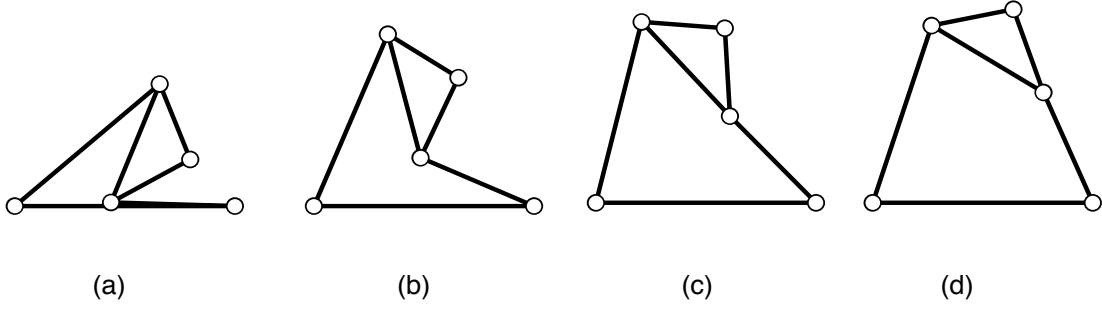


Figure 18: Four snapshots from the lifetime of a 1dof mechanism: (a) and (c) are the limit situations of (b), which is expansive. (d) is not expansive: some diagonals increase, some decrease. Notice the change in the combinatorial structure from (b) to (d), via (c).

Farkas' Lemma for self-stresses and infinitesimal increases. In section 3.1 we defined the four linear spaces associated to the rigidity matrix and stated two orthogonality conditions (6 and 7). The orthogonality condition (6) between the space of self-stresses $\ker M^T$ and the space of infinitesimal length increases $\text{Im } M$ implies that:

$$\sum_{ij \in E} w_{ij} \langle p_i - p_j, v_i - v_j \rangle = 0, \quad \forall w = (w_{ij}) \in \ker M^T, \quad \forall v \in \mathcal{R}^{2n} \quad (9)$$

Let us rewrite it in terms of the values $d_{ij} = \langle p_i - p_j, v_i - v_j \rangle$, the infinitesimal increases.

$$\sum_{ij \in E} w_{ij} d_{ij} = 0, \quad \forall w = (w_{ij}) \in \ker M^T, \quad \forall d_{ij} = \langle p_i - p_j, v_i - v_j \rangle \quad (10)$$

The sum cannot be zero if all the summands are strictly positive. In particular, this implies that there must exist a pair of edges $ij, kl \in E$ such that the signs are different: $w_{ij} d_{ij} > 0 \Leftrightarrow w_{kl} d_{kl} < 0$. We will refer to the sign of w_{ij} as the *self-stress sign*, and to the sign of d_{ij} as the *increase sign*.

Interpreting this condition for all edges proves the following Theorem. Alternatively, this follows from Farkas' Lemma, or Linear Programming duality, see [14].

Theorem 4.8 (Farkas' Lemma for self-stresses) *Let $G(\mathcal{P})$ be a framework, $w \in \ker M^T$ a self-stress on it and v an infinitesimal motion vector which preserves infinitesimally the lengths of some of the edges in G and increases or decreases the others. Denote by $d_{ij} := \langle v_i - v_j, p_i - p_j \rangle$, for $ij \in E$, the infinitesimal increase or decrease value.*

Then there exist at least two edges ij and kl which are not moved rigidly by v , and on which exactly one of the following conditions holds:

1. *Either the self-stress signs on both the edges are the same, $w_{ij} w_{kl} > 0$, and the increase signs differ $d_{ij} d_{kl} < 0$*
2. *Or the self-stress signs differ $w_{ij} w_{kl} < 0$, and the increase signs are the same $d_{ij} d_{kl} > 0$*

We will use the following corollary to prove the expansiveness of pseudo-triangulation mechanisms.

Corollary 4.9 *Let $G(\mathcal{P})$ be a Laman framework with an extra edge and let v be an infinitesimal motion v which acts rigidly on a subset of $2n - 4$ edges of G and which changes infinitesimally the lengths of the other two remaining edges. Then $G(\mathcal{P})$ supports a self-stress, and the signs of the self-stress on the two non rigid edges satisfies exactly one of the following two conditions:*

1. *Either both self-stresses have the same sign, and the two edge increases have different signs*
2. *Or the two self-stresses have different signs and the two edge increases have the same sign*

4.4 Definitions: Combinatorial Equivalence of Pseudo-Triangulations

In this section we define what it means for two pseudo-triangulations or two pseudo-triangulation mechanisms to be combinatorially equivalent. We use this concept to characterize the interval of time when a pseudo-triangulation mechanism is expansive, and to identify the moment when it stops being so. Along the way, we define *combinatorial pointed pseudo-triangulations* and mechanisms.

Plane Graphs. A non-crossing embedding of a connected planar graph G partitions the plane into faces (bounded or unbounded), edges and vertices. Their incidences are fully captured by the vertex *rotations*: the ccw circular order of the edges incident to each vertex in the embedding. A *sphere embedding* of a planar graph refers to a choice of a system of rotations (and thus of a facial structure), and is oblivious of an outer face. A *plane graph* is a spherical graph with a choice of a particular face as the outer face.

A (combinatorial) *angle* (incident to a vertex or a face in a plane graph) is a pair of consecutive edges (consecutive in the order given by the rotations) incident to the vertex or face.

Combinatorial frameworks. A *combinatorial framework* $G(M)$ associated to a framework realization $G(\mathcal{P})$ is obtained by retaining (in M) only some combinatorial information from the underlying oriented matroid M of the set of points $\mathcal{P}$. Since in this paper we work only with special types of frameworks, to keep the focus on the main problem we do not give here the general definition, but see the upcoming paper [74]. In our particular case, when the framework is planar, the information M retained from the embedding will be, for each vertex i , the *signed* circular sequence $i : j_1 \cdots j_{n-1}$ of vertex indices $j \neq i$ in which a *directed line* through p_i , rotating ccw around the vertex encounters the adjacent edge vectors $\overrightarrow{p_i p_j}$. An index j corresponding to the edge vector $\overrightarrow{p_i p_j}$ is recorded positively or negatively depending on whether the rotating directed line encounters it in a way that matches its direction or the opposite one. This concept is a specialization of the *local sequences* of Goodman and Pollack [37] or *hyperline sequences* of Bokowski [22] (see also [72] and [23]) and retains (partial) oriented matroid structure from the underlying set of points $\mathcal{P}$. In particular, we can read off from this information the planar nature of an embedding of the framework, the underlying plane graph (rotations) and the pointed or non-pointed nature of the edge vectors at each vertex.

Combinatorial equivalence of planar frameworks. Two planar frameworks are *combinatorially equivalent* if there is a one-to-one correspondence between their vertices preserving edges and faces, the outer face and its orientation, and the underlying partial oriented matroid, i.e.

the circular ccw order of directed lines through the edge vectors around each vertex. This just means that they have the same underlying combinatorial framework (and the embeddings have the same orientation.)

In particular, this defines *combinatorially equivalent pseudo-triangulations*. The underlying combinatorial framework captures the information. We define now an alternative, equivalent but somewhat easier to understand concept, first described and used in [41].

Combinatorial pseudo-triangulation. Let G be a plane connected graph¹. A *combinatorial pointed pseudo-triangulation (cppt)* of G is an assignment of labels *big* (or *reflex*) and *small* (or *convex*) to the angles of G such that:

- (i) Every face except the outer face gets *three* vertices marked *small*. These will be called the *corners* of the face.
- (ii) The outer face gets only *big* labels (has no corners).
- (iii) Each vertex is incident to exactly one angle labeled *big* and is called *pointed*.

Let $G(\mathcal{P})$ be a pointed pseudo-triangulation (ppt), to which we will associate naturally a combinatorial pointed pseudo-triangulation (cppt) $G(C)$ by forgetting the point coordinates $\mathcal{P}$ and retaining (in C) only the plane graph structure and the label (convex or reflex) of each angle.

Two pointed pseudo-triangulations are *combinatorially equivalent* if they have the same underlying cppt structure.

A fundamental idea of our convexification algorithm described in Section 6 will be to use mechanisms obtained from pointed pseudo-triangulations, and to reconfigure them continuously as long as the combinatorial pseudo-triangulation does not change (where the removed convex hull edge is taken into account when considering the cppt structure). See Figure 18 for an illustration.

4.5 Pointed Pseudo-Triangulation Mechanisms Move Expansively

We are now ready to prove the main rigidity-theoretic result concerning pointed pseudo-triangulation mechanisms: they are expansive 1dof mechanisms for as long as the combinatorial pointed pseudo-triangulation is unchanged. At one of the two endpoints of the expansiveness interval, the configuration space of a ppt-mechanism may sometimes be singular. We show that for certain special situations (which will be the only ones relevant for the convexification algorithm of simple planar polygons), singularities never occur. This will guarantee that the algorithm behaves correctly at those moments.

As a corollary, we obtain that all frameworks which are Laman-minus-one equivalent to a pointed pseudo-triangulation mechanism are also expansive.

Theorem 4.10 (Infinitesimal expansiveness of pseudo-triangulation mechanisms)
A bar-and-joint framework whose underlying graph is obtained by removing a convex hull edge from a pseudo-triangulation is an infinitesimal 1dof expansive mechanism.

¹The definition is valid in a more general setting than what we need in this paper, see [41, 61]

Proof: Let G be the underlying graph of the pointed pseudo-triangulation, e_c be the removed convex hull edge (so that $G \setminus \{e_c\}$ is a ppt-mechanism) and e be an arbitrary edge not in G . As a Laman-minus-one graph, $G \setminus \{e_c\}$ has a natural decomposition into rigid components. We must prove that, when the removed convex hull edge e_c is expanded, so does any other free diagonal e (i.e. one which is not part of a rigid component). Indeed, the *missing diagonals* inside rigid components are not changing in length.

Let v be an infinitesimal motion preserving the edge lengths of $G \setminus \{e_c\}$ and increasing the length of the edge e_c , $d_{e_c} > 0$. We want to prove that it also increases the length of the edge e , i.e. $d_e > 0$. Consider the graph $G \cup \{e\}$, with $2n - 2$ edges. It has a non-trivial self-stress. Corollary 4.9 implies that it is sufficient to prove that this self-stress cannot be strictly positive on both e_c and e . Assume for the sake of a contradiction that this happens, i.e. that $G \cup \{e\}$ has a self-stress w with $w_{e_c} > 0$ and $w_e > 0$. We will interpret this in terms of the induced Maxwell lifting to get a contradiction.

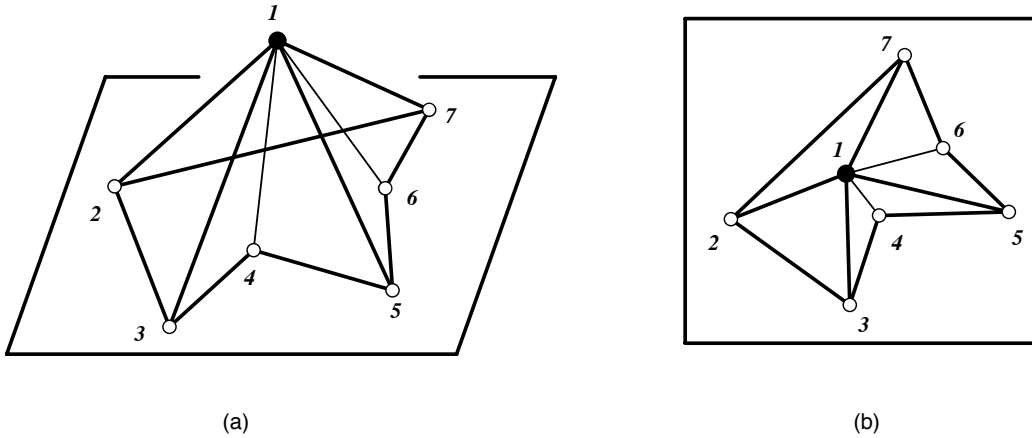


Figure 19: The mountain/valley argument: (a) Cutting just below the vertex of maximum z -coordinate (vertex 1), we get the image in (b). There must be at least 3 mountain edges incident to 1, and their projections are non-pointed. In (a), the mountain edges are 12, 13, 15 and 17.

Consider the framework $G \cup \{e\}$ obtained from G by adding the extra edge e . Since it is no longer a pointed pseudo-triangulation, either an endpoint of this new edge e is non-pointed, or the edge e crosses some other edges of the pseudo-triangulation, or both. If new crossings have been introduced, we will apply Bow's construction to obtain a new planar framework $G'(\mathcal{P}')$, which will have n' vertices and $2n' - 2$ edges. In either case, the new framework is non-pointed only at (one or possibly both of) the endpoints of e or at the crossings of e with other edges. Denote by M the set of non-pointed vertices: the endpoints of e , if non-pointed, and the crossings of e with other edges, if there are such crossings. M is non empty.

The signs of the self-stresses are preserved by Bow's construction. Since we assumed a strictly positive self-stress on e_c and e , this means that both e_c and e are valley edges, in a Maxwell lifting of $G \cup \{e\}$ (and this extends to the split edges in case the Bow's construction has been

applied). It means that the only edges which could be mountain edges are the edges of G .

We now apply the same argument as in Theorem 3.7. We look for a vertex of maximum height and conclude that it must be non-pointed, hence only those in M are candidates.

We show first that the maximum cannot be attained (locally) on more than a vertex, i.e. not on an edge or face or larger substructure. Indeed, if an edge would be at maximum height, its two endpoints would have to be in M . But the edges between vertices in M are either e or splittings of e and have negative stress, therefore lift to valleys. Valleys cannot be maxima, obviously.

To complete the proof, let's focus on the vertex of maximum height and call it the *tip*. If the tip is a Bow vertex, then its four incident edges must all be mountain edges, because of their collinearity. These include the split added edge e . But this contradicts the previous argument that only edges of G may be mountain edges.

So we may now assume without loss of generality that the maximum is vertex 1 (see Fig. 19). Because 1 is an isolated maximum (not part of a local maximum subgraph), then all its incident faces are slanted (non-horizontal). Cutting the lifted polyhedral surface with a plane slightly below the tip, we get in this cut plane a simple closed polygon (therefore, with at least three vertices) containing the projection of the tip vertex 1 inside it, and therefore inside its convex hull. The projections on the cut plane of the polyhedral surface edges incident to 1 which are *mountain edges* must join the projection of the tip to the convex hull vertices of the projection in a non-pointed manner (because the projection lies in their convex hull). See Fig. 19 for an illustration. But these are exactly the edges incident to 1 in G , and they form a pointed edge set, hence the contradiction. $\square$

Corollary 4.11 *Let $G(\mathcal{P})$ and $G'(\mathcal{P})$ be generic realizations of equivalent Laman-minus-one graphs. If $G(\mathcal{P})$ is a pointed pseudo-triangulation mechanism, then $G'(\mathcal{P})$ is also a 1dof expansive mechanism.*

Proof: We replace an infinitesimally rigid component by another one with the same property. The space of infinitesimal motions and the expansion pattern do not change. $\square$

Flexible expansiveness of pointed pseudo-triangulation mechanisms. We summarize what we have shown so far.

A bar-and-joint framework $G(\mathcal{P})$ obtained by removing a convex hull edge e from a pseudo-triangulation is a flexible 1dof expansive mechanism for as long as the combinatorial pseudo-triangulation (of $G \cup \{e\}$) doesn't change. The interval of expansiveness is bounded by two special embeddings of $G \cup \{e\}$: one has (at least one) semi-simple pseudo-triangular face (with at least one zero-angle corner), the other has (at least one) aligned pair of extreme edges (an angle of π at a non-corner). A special instance of the latter case occurs when the missing convex hull edge aligns with another convex hull edge.

The interval contains no singular points (of the configuration space), since pseudo-triangulations are infinitesimally rigid. Hence the mechanism is smooth in this interval. It remains to prove (in the next section) that if the semi-simple pseudo-triangular face is not collapsed to a flat polygon, then this extreme embedding is not singular. The other extreme embedding (with an aligned vertex) is never a singularity.

4.6 Pseudo-triangulation mechanisms with semi-simple but not flat pseudo-triangles are non singular

There is only one important thing left to be proven: that singularities cannot occur, for the types of pointed pseudo-triangulation mechanisms used in this paper. More precisely, we have to address the case of those pointed pseudo-triangulations which contain semi simple, but not flat pseudo-triangles. A *flat* pseudo-triangle has all its vertices embedded on a line, and is therefore singular. But such flat pseudo-triangles never appear in the convexification algorithm described in Section 6. The algorithm uses, however, semi-simple pseudo-triangles: these can have some flat corners (possibly more than one), but not be altogether flat. Moreover, even when zero-angle corners occur, the incident edges do not overlap completely. Indeed, this is the definition of semi-simplicity given in Section 2.1.

We show now that these *semi simple pointed pseudo-triangulations* satisfy the same properties as pointed pseudo-triangulations: they are self-stress-free. With a convex hull edge is removed, such an object will appear as the starting point of the expansion interval of the underlying graph. By proving that they are never stressed, we guarantee that the configuration space is smooth at that point, which implies that the motion is well defined.

Theorem 4.12 *Semi simple pseudo-triangulations are infinitesimally rigid.*

Proof: We need only a simple extension of the argument used in the proof of Theorem 3.7, to handle the semi simple pseudo-triangular faces. If we assume that there exists a lifting where the two overlapping edges are lifted to two distinct lines, then there is no way of obtaining a lifted flat face for the semi-simple pseudo-triangle. So the two edges must overlap in the lifting. But then, the face that must be lifted appears in the lifting as a (pure) pseudo-triangle (plus a segment attached to it), with a new corner at the position where the endpoint of one of the two overlapping edges lies on the other edge. Extending the argument used in Theorem 3.7, the only possible placement of a vertex of maximal height would then be at this new corner. But this is impossible, because it belongs to a lifted edge (the larger of the two overlapping ones). The whole edge cannot be at maximum height, because its endpoints are pointed. Hence the contradiction to the existence of a lifting, and of a self-stress. $\square$

Corollary 4.13 *Semi simple pseudo-triangulations with a convex hull removed are self-stress-free, hence non-singular.*

Proof: Removing an edge from a Laman graph in a generic embedding results in a self-stress-free Laman-minus-one graph. $\square$

This concludes the sequence of rigidity theoretic properties of pointed pseudo-triangulations and allows us to use them as building blocks describing simple motions for the convexification algorithm. We summarize their properties in the following Theorem, on which the correctness of the Algorithm described in Section 6 relies.

Theorem 4.14 (Motion of Pointed Pseudo-Triangulation Mechanisms) *A pointed pseudo-triangulation mechanism moves expansively on a unique, well defined (one-dimensional) trajectory in its configuration space, from the moment when it is embedded as a semi-simple pseudo-triangulation to the moment when two extreme edges of a vertex (or, as a special case, when one of these is the missing convex hull edge) align.*

Proof: The statement about the beginning of the expansion interval follows from Corollary 4.13. At the other endpoint, since all distances increase, so do all the convex angles. The pointed pseudo-triangulation remains the same, combinatorially, until two extreme edges incident to a vertex align. Notice that one of these two aligned edges may be the missing convex hull edge. $\square$

5 Algorithms for Constructing Pseudo-Triangulations

In this section we investigate several algorithms for constructing pointed pseudo-triangulations of planar point sets and polygons. They rely on the maximality property 5 from Theorem 2.3, which implies that we can extend any non-crossing and pointed edge set to a pointed pseudo-triangulation by adding edges in an arbitrary order, as long as we do not violate these two properties (pointedness and non-crossing). In particular, any simple polygon (which is a pointed and non-crossing edge set) can be extended to a pointed pseudo-triangulation.

5.1 Pseudo-triangulations of planar point sets

The input to a pseudo-triangulation algorithm is a planar point set, not all collinear. The result must be a (possibly semi-simple) pointed pseudo-triangulation. For simplicity, we will assume general position (no three points are collinear). Some of the algorithms we will describe would work without major modifications under less restrictive conditions.

All throughout, we will assume that the pointed pseudo-triangulation is stored as a planar map, e.g. using the quad-edge data structure of [39].

A pseudo-triangulation is called *Henneberg I*, if it can be constructed inductively using only Henneberg I steps, cf. Theorem 2.3(6).

Incremental algorithm. This algorithm assumes that the points are in general position and adds the points one at a time. It produces Henneberg I pseudo-triangulations.

Algorithm 5.1 (Incremental Algorithm)

Input: A point set $\mathcal{P}$.

Output: A pointed pseudo-triangulation of $\mathcal{P}$.

1. Start with three arbitrary points, join them in a triangle.
2. Add the remaining points one at a time. Each new point falls inside a face of the previously constructed pointed pseudo-triangulation. Take two tangents to the side chains of the pseudo-triangular face to extend the pseudo-triangulation. Readjust the planar graph data structure.

The complexity of this algorithm is as follows. There are n steps. Each step requires a point location phase, to find the face containing the new point ($O(\log n)$ time), a tangent finding phase (linear time in the size of the face) and an insertion phase, with an update of the planar map data structure. A rough analysis gives a worst case running time of $O(n^2)$.

Sweep line algorithm. Add points in increasing order of their x -coordinates, or in any sweep line order. It is a simpler version of the previous algorithm (the incremental one) and produces pseudo-triangulations that are constructed using only Henneberg I steps on the outer face.

Algorithm 5.2 (Sweep line Algorithm)**Input:** *A point set $\mathcal{P}$.***Output:** *A pointed pseudo-triangulation of $\mathcal{P}$*

1. *Sort the points by x coordinate.*
2. *Make a triangle from the first three points.*
3. *Add the remaining points in increasing order of their x -coordinates. At each step, take two tangents to the convex hull of the already constructed pseudo-triangulation.*

The complexity of this algorithm is $O(n \log n)$. The initial sorting takes that much. The rest of the algorithm is identical to the incremental algorithm for constructing the convex hull of a point set, except that here we retain all the intermediate tangents. The amortized analysis for the convex hull algorithm gives linear time total for all the work after the initial sorting phase.

Greedy flip algorithm. The previous algorithms produce only Henneberg I pseudo-triangulations. We may flip some diagonals to obtain other types of pseudo-triangulations. A well developed algorithm is described in Pocchiola and Vegter [63].

Algorithm 5.3 (Greedy Flip Algorithm [63])**Input:** *A point set $\mathcal{P}$.***Output:** *A pointed pseudo-triangulation of $\mathcal{P}$.*

1. *Start with a sweep line pseudo-triangulation.*
2. *Flip some edges in a greedy manner: the smallest slope first.*

Complexity. Because the convex hull is part of any pseudo-triangulation of a point set, the well known lower bound of $\Omega(n \log n)$ applies, and hence we cannot do any better than the sweep line algorithm.

5.2 Pseudo-triangulations of simple polygons

We expect to be able to beat $O(n \log n)$ when constructing a pseudo-triangulation of a polygon. Indeed, we show that this is possible if we rely on the linear time polygon triangulation algorithm of Chazelle [28]. For a practical implementation, we also describe some other alternative approaches (or we rely on other sub-optimal polygon triangulation algorithms).

Incremental Algorithm for polygons. The pseudo-triangulation is constructed incrementally, adding one edge of the polygon at a time, in ccw order, starting with a vertex on the convex hull. This insures that the last step of the algorithm will reuse a previously inserted edge and won't necessitate any additional deletions. At each step of the insertion we add a vertex, a polygon edge and one other additional edge. However, we might have to displace or modify several other edges, depending on whether the pointedness condition, the planarity condition, or both are violated by the insertion of the new polygon edge.

We show how to modify some of the added pseudo-triangulation edges using an argument which we call the *rubber band argument with snapping*, which locally modifies (in worst case linear time) the edges adjacent to the new polygon edge to preserve pointedness and planarity. The resulting worst case complexity is $O(n^2)$ time.

Algorithm 5.4 (Incremental Algorithm for Polygons)**Input:** A simple polygon $\mathcal{P}$ given by its point set, in ccw order around its boundary.**Output:** A pointed pseudo-triangulation of $\mathcal{P}$

1. Start from a vertex on the convex hull of the polygon and the first two edges to obtain a triangle.
2. Add the vertices one at a time, together with a new polygon edge.
3. If pointedness is violated, split the existing edges incident to the last vertex into two parts, using the new edge as a divider. Move half of them (the part containing only added edges, not the other polygon edge incident to the vertex) to the new endpoint. Recurse, to maintain planarity and pointedness.
4. If planarity is violated, it is because the new edge is cutting through several added edges. Subdivide these crossed edges into two, and recurse to maintain planarity and pointedness on each half. Some of these new edges will coincide.

See Figure 20 for an example. The correctness proof would show that at step 4, only one new edge is inserted. We omit the details of the implementation and analysis, because we won't rely on this approach.

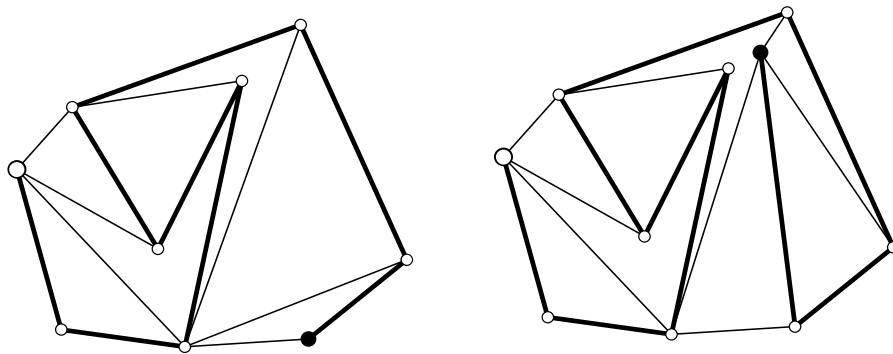


Figure 20: A typical step, violating planarity, of the incremental algorithm for computing the pseudo-triangulation of a polygon. The first vertex is white and big, and the last inserted vertex is black. The polygon edges are thick and the new diagonals are thin.

Shortest Path (Geodesic) Tree algorithm. The *reduced shortest path tree* inside a polygon, rooted at a vertex i , computes the shortest paths from a convex hull vertex i to all the other vertices, such that adding the edges of the path doesn't produce (at its endpoint) a non-pointed vertex. For this to happen, the destination of the path must be a reflex vertex of the polygon. It suffices to compute all the paths ending at convex vertices. The collection of edges of the reduced shortest path tree is planar and pointed. If we compute it for the polygon and all its pockets

(determined by the complement of the polygon with respect to its convex hull), we obtain a pointed pseudo-triangulation of the point set of the polygon, which includes the polygon edges.

All the steps of this algorithm have already been described in the literature. To compute the convex hull of the polygon and identify its pockets takes linear time. We triangulate in linear time using Chazelle's [28] algorithm. Inside the polygon and in each of the pockets, a shortest path tree from a vertex using the triangulation, is done in linear time using the algorithm in [40]. We retain only the paths leading to convex vertices.

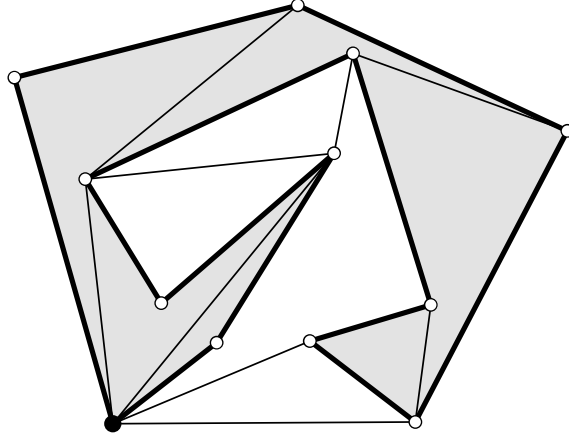


Figure 21: The shortest path tree pseudo-triangulation of a polygon. The polygon edges are thick and the new diagonals are thin. The root of the tree, for the polygon and its unique pocket, is black.

6 The Convexification Algorithm

We are now ready to present the main results of the paper, Algorithm 6.1 for convexifying a simple planar polygon with expansive motions, and its correctness proof, Theorem 6.4.

To convexify a polygon, our approach is to first construct a pointed pseudo-triangulation, then turn it into an expansive mechanism by removing a convex hull edge, and then move it until the end of its expansive interval. At that point we apply a local flip to obtain another pseudo-triangulation mechanism, and continue like this until reaching a convex position.

Reconfigurable frameworks. Two realizations $G(\mathcal{P})$ and $G(\mathcal{P}')$ of the same framework (G, L) are said to be *reconfigurable* into one another, if they belong to the same component of the realization space. We want to reconfigure two simple polygonal frameworks, without producing self-intersections along the way. For this, the two polygons must be similarly oriented.

Reconfiguring more general frameworks. The algorithm can be applied to collections of open and closed polygons, none inside the other, or to more general collections of objects consisting in a pointed and non-crossing collection of edges. In this last case, the expansive,

non-colliding motion will continue only as long as planarity and pointedness are maintained by the ability to flip an added edge (see below).

6.1 The Convexification Algorithm

The main algorithmic result of the paper can now be described.

Algorithm 6.1 (The Pseudo-Triangulation Convexification Algorithm)

Input: *A simple planar polygon on the point set $\mathcal{P}$.*

Output: *A convexification of $\mathcal{P}$, and a sequence of motions, each one given by a pseudo-triangulation mechanism, describing the convexification process.*

1. *Compute an initial pseudo-triangulation of $\mathcal{P}$, and remove an arbitrary convex hull edge which is not a polygon edge.*
2. *Repeat until a convex position is reached:*
 - (a) *Pin down an arbitrary edge. Move the pseudo-triangulation mechanism along its unique trajectory, in the expansive direction, until an alignment event occurs.*
 - (b) *At the alignment event, perform either a flip or a freeze operation (defined below). If necessary, reconfigure locally the pseudo-triangulation (see below how), and continue.*

See the examples from Figures 1, 2 and 3.

In the rest of this section, we give the missing definitions, the correctness proof and the time analysis.

6.2 Alignment Events

Each expansive mechanism induced by a pseudo-triangulation can be moved as long as the framework stays pointed. Pointedness is violated when two extreme edges at a vertex align. We now show how to reconfigure the mechanism and continue the motion after an alignment event. The overall motion is obtained by gluing together the trajectories corresponding to the motions happening between two events.

When two bars become collinear, we have to recompute a new pseudo-triangulation. Theorem 6.2 proves that this can be done either with local changes and keeping the same number of vertices in the original polygon, or by decreasing by at least one the number of vertices of the polygon and applying induction.

Theorem 6.2 (Gluing trajectories at alignment events) *When two edges align, one of the following three cases may occur:*

1. **Freeze event.** *Two adjacent edges of the polygon become collinear. In this case, we freeze the joint, eliminating one vertex of the polygon.*
2. **Flip event.** *Two adjacent added diagonals or one diagonal and an edge of the polygon become collinear. In this case, we perform a flip in the pseudo-triangulation to obtain a pseudo-triangulation with a semi simple face.*

3. **Convex hull false event.** *The missing convex hull edge and either a diagonal or a polygonal edge align. In this case, we continue the motion and do not consider this a proper event. Note that the missing edge will change combinatorially, as does the convex hull.*

An example is depicted in Fig. 22.

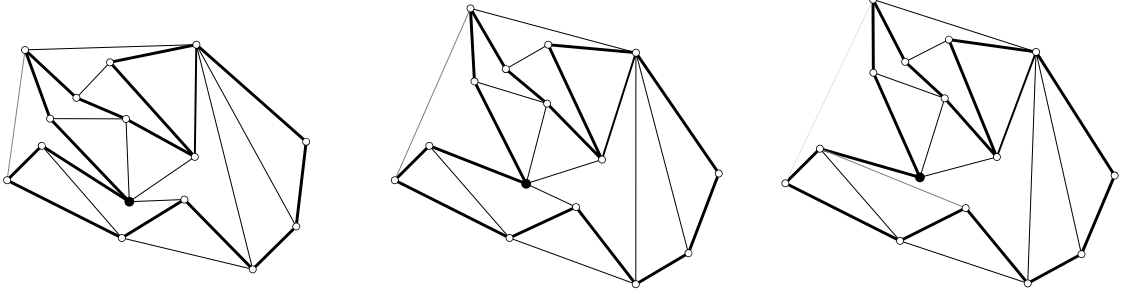


Figure 22: Patching the pseudo-triangulation by a local flip when two bars (not both of which are polygon edges) align.

The above theorem is not stated in full generality, to avoid cluttering the overall picture with details. In particular, several vertices may straighten simultaneously, but the same type of flipping would work in each case.

Other problems occur when we freeze two aligned edges of the polygon. In this case we must get rid of the other diagonals (if any) adjacent to the vertex, which can also be done by local changes (but may involve linearly many edges). Occasionally this operation rigidifies the framework: then we must pick up another convex hull edge to remove from the convex hull. All these details are solved in a straightforward manner.

Theorem 6.3 (Termination and complexity analysis) *The convexification of a polygon terminates in at most $\mathcal{O}(n^3)$ steps, where a step is the continuous motion induced by one pseudo-triangulation mechanism, between two alignment (flip or freeze) events.*

Proof: With some care in the patching strategy, it can be shown that no combinatorial pseudo-triangulation will occur twice in the convexifying motion. Since there are finitely many combinatorial pseudo-triangulations, the algorithm terminates.

A more careful accounting refines the analysis and gives at most $\mathcal{O}(n^3)$ steps. We do so by bounding the number of flip events between freeze events, which are exactly the number $r = \mathcal{O}(n)$ of reflex vertices in the polygon.

Consider the number of bends in all the shortest paths forming the pseudo-triangulation, inside the polygon and its pockets. Since there are linearly many shortest paths accounted for (we consider maximal paths only, not subpaths), and each can have at most linearly many bends, the crude bound on this number is $\mathcal{O}(n^2)$. In certain situations, the bound may even be attained, for instance when an added diagonal flips along the edges of a big $\mathcal{O}(n)$ -sized rigid (and therefore convex) component. The key observation is that a local flip decreases the number

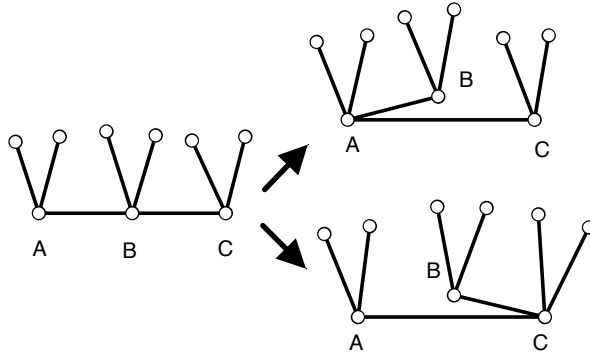


Figure 23: When the two aligned edges are not polygon edges, there are two possible diagonal flips. A flip reduces the number of bends in geodesic paths using the vertex involved in the alignment.

of bends by at least one (or more, depending on how many geodesic paths go through the vertex where the event happened). Hence the number of steps is at most $\mathcal{O}(rn^2) = \mathcal{O}(n^3)$. $\square$

The main result is now a consequence of all the results proven so far:

Theorem 6.4 (Main Result: Convexification of Planar Chains with Expansive Motions) *Every planar polygon can be convexified with at most $\mathcal{O}(n^3)$ motions. Each motion is induced by a 1dof expansive mechanism constructed from a pseudo-triangulation with a deleted hull edge, which is continuously reconfigured expansively until two adjacent edges align. At that point a local flip of the diagonals restores a pseudo-triangulation. The complete trajectory in configuration space is a sequence of simple curves, each one naturally parametrized by a rotating edge in the plane. The first pseudo-triangulation can be computed efficiently in linear time and updated in (at most) linear time per step.*

7 Conclusions

More than just giving a particular solution to the Carpenter’s Rule problem, our combinatorial approach to planning non-colliding motions based on pseudo-triangulations has several conceptual consequences. We conclude with a brief discussion of further directions of research and remaining open questions or applications to be considered, as well as related papers, resources and recent work on pseudo-triangulations and their applications.

Note on multiple papers. While working on the full version of [73], I realized that it would be too much material for a single self-contained paper. I decided to split it into three parts, of which the current paper should be read first.

The second part, describing some of the numerical issues of the algorithm implementation, has appeared in [75]. Here’s a short summary of these considerations. The algorithm we presented works in steps, but from the configuration in the beginning of a step to the one at the end, there is no constant-time algebraic computation. Indeed, finding the coordinates of the points at

the alignment events involves solving a system of quadratic equations in $2n - 3$ equations with that many unknowns. Solving them exactly may lead, in the worst case, to exponentially many real solutions (see [24]), from which the solution corresponding to the actual position during the motion would have to be isolated based on combinatorial considerations. Moreover, there is no simple algorithm for finding all of these solutions, as elimination-based solvers work in exponential time. In practice, the systems are solved numerically. See also the Note below on the model of computation and complexity.

The third part [74] (in preparation) will deal with the observation (and its consequences) that certain geometric facts used in the current paper have natural extensions, proofs and applications in the purely combinatorial setting of rank 3 affine oriented matroids (pseudo point sets). We explore, in particular, to what extent is the rigidity matroid *orientable* in a combinatorially recognizable way (based on the oriented matroid of the underlying point set). Indeed, pointed pseudo-triangulation mechanisms induce instances of combinatorially recognizable signed co-circuits. In particular, certain simple geometric facts used in the current paper can be precisely derived in this context.

Note on terminology. In the conference version of this paper, I used *acyclic* instead of *pointed*. The terminology of *acyclic vector sets* comes from oriented matroid theory (see [21]), an approach which provided the guiding line in the search for combinatorial properties of simple motions induced by mechanisms. More precisely, for a set of acyclic vectors there is no linear combination with positive, not all zero coefficients that sums them to zero, while in the cyclic case there is always one. The decision to change the terminology was motivated by the obvious potential of confusion with the entrenched graph-theoretic concept of acyclic graph.

Note on the Model of Computation and Complexity. Computationally, our problem has as input an ordered list of $2n$ real numbers, the coordinates of the n vertices of the polygon. The desired output is a finite description of the convexifying motion, and ultimately a set of $2n$ real coordinates for the polygon vertices in the final convex state.

The moving vertices follow continuous curve segments during this motion, which can be parametrized by time. Ideally, one would like to express all the computations leading to parametric descriptions of these $n - 2$ vertex trajectories in the real RAM model, the computational model of choice in Computational Geometry (see [64]). However, as is the case for other algorithms in Real Algebraic Geometry (see [17]), not all numerical data can be computed explicitly. The positions of the polygon vertices at the event points are given implicitly as solutions to systems of polynomial equations. Theoretically, one may attempt to perform the costly (exponential time) exact algebraic elimination (Gröbner basis algorithms) to reduce the system to a single one-variable polynomial equation, but then one still would have to find the roots of this polynomial numerically (i.e. not in the RAM model). In practice, and since we must rely on numerical methods anyway for root finding, everything is computed approximately via numerical methods, see [75].

In [73], the number of steps in the convexification algorithm was hastily estimated as $O(n^2)$ instead of $O(n^3)$.

Animations and web resources. A web page (constructed with the help of my students Elif Tosun and Beenish Chaudry) containing animations and other graphical illustrations of the approach described in this paper, can be found on the author's web page at:

<http://cs.smith.edu/~streinu/Research/Motion/motion.html>

7.1 Open Questions on Reconfiguring Chains

Several questions directly connected to our approach remain still open at the time of this writing.

In practical implementations of our algorithm, we have observed that the number of events is often linear. We can easily produce an example where a shortest-path pseudo-triangulation approach would encounter a quadratic number of flip-events between two freeze events. Take a pseudo-triangulation containing a rigid (and hence convex) component with linearly many vertices, and design shortest paths from a vertex to linearly many other vertices, which are obstructed by the large convex component and must bend around it. It is not hard to design a pseudo-triangulation mechanism where all these $O(n)$ paths would unfold one by one, for a total of $O(n^2)$ steps, before any freeze event would happen. It is also easy to design a freeze event which would readjust the pseudo-triangulation by introducing all these bends again. However, neither the shortest path tree pseudo-triangulation, nor the patching strategy for the pseudo-triangulation (at a freeze event) are the only available options. It is possible that other choices of pseudo-triangulations would lead to faster convexification algorithms.

Open Problem 1 *Is there a different strategy of choosing pseudo-triangulation mechanisms, leading to asymptotically fewer events?*

Another potential source of simplification is in the complexity of the flip operation.

Open Problem 2 *Is it possible to perform a constant time flip at an alignment event, while maintaining an overall small number of events during the convexification algorithm?*

The most intriguing remaining question is whether the approach based on expansive motions and pseudo-triangulations can be extended to non-crossing polygons which are not necessarily simple. To define the problem, let's introduce some additional terminology. Related problems, and more general self-touching linkages have also been recently considered in [31].

Two edges are non-crossing if their segments do not have an interior point in common, otherwise they are said to cross properly. Non-crossing edges may be touching, either overlapping, or having an endpoint of one edge lying on another edge. We view each edge as oriented and having two "sides" (left or right, with respect to a given direction), so that when two edges are touching or overlapping, we can specify on which side they meet. For example, we think of the two edges 12 and 45 in Fig. 24 (oriented from 1 to 2, and from 4 to 5) as touching on the left of both 12 and 45.

A *non-crossing polygon* has no properly crossing edges, although the edges may be touching in complicated ways. Therefore, we also need to add to the definition of the polygon a *touching pattern*, defined in such a way that would allow some simple, non-crossing unfolding reconfiguration (not necessarily maintaining rigidly the lengths of the edges). The touching pattern indicates which side of each edge is in contact with which side of another edge, in a consistent fashion. A touching pattern is *consistent* if there exists a (topological) simple realization of the polygon (not necessarily with straight-line segments, but simple, i.e. non-touching), which is consistent with the sidedness relation.

For instance, if the touching pattern of the two edges 12 and 45 in Fig. 24 was defined such that 12 was in contact on its right side with the right side of edge 45, then that wouldn't have been considered a non-crossing touching pattern, as the two touching edges would have had to slide over each other to get "on the other side", in order to become a simple polygon.

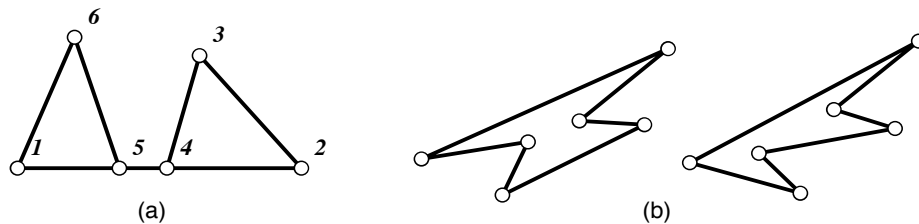


Figure 24: (a) A non-crossing, but not simple polygon. (b) Two possible polygonal patterns (given by the above/below relationships between edges) compatible with the same flat polygon.

Open Problem 3 *Is it possible to convexify every non-crossing polygon (with any consistent touching pattern)?*

The special situation when all the edges are collinear is the only case when the configuration space of a polygon has singularities, see [45]. This creates additional questions, regarding which are the combinatorial possibilities (given by touching patterns) for an opening flat polygon. For the simple example of a four-gon, see Fig. 17. Figure 24 (b) suggests two possible opening patterns of a flat six-gon.

Open Problem 4 *Given a flat n -gon and a consistent touching pattern, is there a configuration, infinitesimally close to the flat singular position, which opens the polygon in a manner compatible with the given pattern?*

Open Problem 5 *Are all the consistent simple touching patterns of a flat polygon realizable by infinitesimally close configurations (with the same edge lengths)?*

Related questions refer to finding (possibly semi-algebraic) conditions that would distinguish between opening motions along trajectories with distinct combinatorial patterns, when moving away from a singularity point.

7.2 Recent work and Open Questions on Pseudo-Triangulations

Since the conference abstract was published [73], several questions regarding the combinatorial properties of pseudo-triangulations have been addressed. Without making a claim at giving a complete listing, we refer here to some of these developments, as well as other interesting remaining open questions.

Open Problem 6 *How many pseudo-triangulations of a point set or polygon are there?*

Partial results are given in [65, 8, 6]. A related question is to enumerate efficiently all pseudo-triangulations. Current approaches include [18, 26, 7]. A remaining questions is:

Open Problem 7 *Find a simple algorithm to enumerate pseudo-triangulations with constant amount of work per new enumerated object.*

More generally, it is natural to ask whether there is an interesting polytope whose 1-skeleton is the graph of pseudo-triangulations. Solutions have appeared for pointed pseudo-triangulations [68] and general (not necessarily pointed) ones, [60], [2], [4]. An interesting remaining question is:

Open Problem 8 *Is there a concept of a regular pseudo-triangulation, as it is for triangulations ([35])?*

Since pseudo-triangulations can be naturally defined in the coordinate-free context of oriented matroids, we are also interested in those combinatorial properties that would distinguish them (as a collection) from the ones realizable in the Euclidean plane. In [41], it was shown that all planar Laman graphs are realizable as pointed pseudo-triangulations, even when an additional combinatorial pseudo-triangular structure was enforced. Additional properties of combinatorial pseudo-triangulations were studied in [61].

Special types of pseudo-triangulations were studied in [47, 5] and [69]. Further applications in visibility [71] and kinetic data structures for collision detection have appeared in [50, 49]. A recent application to single-vertex origami and unfoldings of spherical linkages is given in [77]. Surfaces from pseudo-triangulations and reciprocal diagrams of pseudo-triangulations were studied in [59] and [3].

One of the most intriguing remaining open question is to extend this work in higher dimensions.

Open Problem 9 *Extend combinatorial expansive motions and pointed pseudo-triangulations to three and higher-dimensions.*

Some partial work in the direction of combinatorial conditions for expansive motions in three dimensions is contained in [77].

From the point of view of Rigidity Theory, pointed pseudo-triangulations are a class of planar minimally rigid graphs. There is an analogous theory of rigidity with fixed edge-directions (rather than fixed edge-lengths). In some recent work of the author [76], pointed pseudo-triangulation mechanisms have been shown to possess interesting properties in this model, too.

All the pictures in this paper have been produced using the software package Cinderella [67], which supports motions of 1dof mechanisms and even animates them. But not all pointed pseudo-triangulation mechanisms seem to admit a Cinderella construction.

Open Problem 10 *Prove that not all pointed pseudo-triangulations admit a ruler-and-compass construction. Characterize those which do.*

Lastly, two software implementations of pseudo-triangulations are freely available [25, 46].

Acknowledgements. This work was possible due to the stimulating research environment I encountered at the McGill University Workshops on Folding and Unfolding Problems in Computational Geometry, Barbados 1998 and 2000, the Monte Verita 1999 Discrete and Computational Geometry Workshop in Ascona, Switzerland, and the Geometry Festival, an International Workshop on Discrete Geometry and Rigidity in Budapest, Hungary, Nov. 1999. I am grateful to all the organizers for their invitation (András Bezdek, Károly Bezdek, Károly Böröczy, Bob Connelly, Jacob E. Goodman, Anna Lubiw, Ricky Pollack, Emo Welzl and Sue Whitesides) and to

the many participants (too many to list here, but see [19], [20]) for enthusiastic conversations on this and related problems.

I am particularly indebted to Günter Rote for discussions of his ideas on the use of expansive motions and the linear programming approach to the Carpenter’s Rule problem from [30].

I acknowledge useful conversations on various aspects of Rigidity Theory with Bob Connelly, Henri Crapo, Ruth Haas, Brigitte and Herman Servatius and Walter Whiteley. Many thanks to Ciprian Borcea for pointing several connections with the mathematical literature and for stimulating conversations on aspects of rigidity theory pertaining to real algebraic geometry.

I acknowledge partial NSF support for organizing the Barbados workshop on pseudo-triangulations, Jan. 26-Feb 2, 2001, where some of the open problems raised in the preliminary version of this paper [73] have been solved.

References

- [1] Pankaj Agarwal, Julien Basch, Leonidas Guibas, John Hershberger, and Li Zhang. Deformable free space tilings for kinetic collision detection. *International Journal of Robotics Research*, 2003. Preliminary version appeared in Proc. 4th International Workshop on Algorithmic Foundations of Robotics (WAFR), 2000.
- [2] O. Aichholzer, F. Aurenhammer, P. Brass, and H. Krasser. Pseudo-triangulations from surfaces and a novel type of edge flip. *SIAM Journal on Computing*, 32:1621–1653, 2003.
- [3] O. Aichholzer, F. Aurenhammer, P. Brass, and H. Krasser. Spatial embedding of pseudo-triangulations. In *Proc. 19th Ann. ACM Symp. Computational Geometry*, volume 19, pages 144–153, San Diego, California, USA, 2003.
- [4] O. Aichholzer, F. Aurenhammer, and H. Krasser. Adapting (pseudo)-triangulations with a near-linear number of edge flips. In *Lecture Notes in Computer Science 2748, Proc. 8th International Workshop on Algorithms and Data Structures (WADS)*, volume 2748, pages 12–24, 2003.
- [5] O. Aichholzer, M. Hoffmann, B. Speckmann, and C. D. Tóth. Degree bounds for constrained pseudo-triangulations. In *Proc. 15th Annual Canadian Conference on Computational Geometry CCCG 2003*, pages 155–158, Halifax, Nova Scotia, Canada, 2003.
- [6] O. Aichholzer, D. Orden, F. Santos, and B. Speckmann. On the number of pseudo-triangulations of certain point sets. In *Proc. 15th Annual Canadian Conference on Computational Geometry CCCG 2003*, pages 141–144, Halifax, Nova Scotia, Canada, 2003.
- [7] O. Aichholzer, G. Rote, B. Speckmann, and I. Streinu. The zigzag path of a pseudo-triangulation. In *Lecture Notes in Computer Science 2748, Proc. 8th International Workshop on Algorithms and Data Structures (WADS)*, volume 2748, pages 377–389, 2003.
- [8] Oswin Aichholzer, Franz Aurenhammer, Hannes Krasser, and Bettina Speckmann. Convexity minimizes pseudo-triangulations. In *Proc. 14th Canad. Conf. Comp. Geom.*, 2002. Full version to appear in *Computational Geometry: Theory and Applications*, 2004.

- [9] Oswin Aichholzer, Erik D. Demaine, Jeff Erickson, Ferran Hurtado, Mark Overmars, Michael A. Soss, and Godfried T. Toussaint. Reconfiguring convex polygons. *Computational Geometry: Theory and Applications*, 20(1–2):85–95, October 2001.
- [10] Vladimir I. Arnol’d. *Ordinary Differential Equations*. Springer Verlag, New York, 1992. Translated from Russian, 3rd edition 1984.
- [11] Boris Aronov, Jacob E. Goodman, and Richard Pollack. Convexifying a planar polygon in $\mathcal{R}^3$. Manuscript, October 1999.
- [12] L. Asimow and B. Roth. The rigidity of graphs. *Transactions of the American Mathematical Society*, 245:279–289, 1978.
- [13] L. Asimow and B. Roth. The rigidity of graphs II. *Journal of Mathematical Analysis and Applications*, 68(1):171–190, 1979.
- [14] Achim Bachem and Walter Kern. *Linear Programming Duality, an Introduction to Oriented Matroids*. Springer Verlag, 1992.
- [15] Julien Basch, Jeff Erickson, Leonidas Guibas, John Hersberger, and Li Zhang. Kinetic collision detection between two simple polygons. In *Proc. 10th ACM-SIAM Symp. Discr. Algorithms (SODA)*, pages 102–111, 1999.
- [16] Saugata Basu, Richard Pollack, and Marie-Francoise Roy. Computing roadmaps of semi-algebraic sets on a variety. *Journal AMS*, 13:55–82, 2000. Prelim. version in *Proc. 28th ACM Symp. Theory of Computing*, pp. 168–173, 1996.
- [17] Saugata Basu, Richard Pollack, and Marie-Francoise Roy. *Algorithms in Real Algebraic Geometry*. Algorithms and Computation in Mathematics. Springer Verlag, New York, 2003.
- [18] Sergei Bespamyatnikh. Enumerating pseudo-triangulations in the plane. In *Proc. 14th Canad. Conf. Comp. Geom.*, pages 162–166, 2002. To appear in *Computational Geometry: Theory and Applications*, 2004.
- [19] Therese Biedl, Erik Demaine, Martin Demaine, Sylvain Lazard, Anna Lubiw, Joseph O’Rourke, Steve Robbins, Ileana Streinu, Godfried Toussaint, and Sue Whitesides. Locked and unlocked polygonal chains in three dimensions. *Discrete & Computational Geometry*, 26(3):269–281, October 2001.
- [20] Therese Biedl, Erik Demaine, Martin Demaine, Sylvain Lazard, Anna Lubiw, Joseph O’Rourke, Steve Robbins, Ileana Streinu, Godfried Toussaint, and Sue Whitesides. On reconfiguring tree linkages: Trees can lock. *Discrete Applied Mathematics*, 117:293–297, 2002. Preliminary version appeared in *Proceedings of the 10th Canadian Conference on Computational Geometry*, abs:4–5, 1998.
- [21] Anders Björner, Michel Las Vergnas, Bernd Sturmfels, Neil White, and Günter Ziegler. *Oriented Matroids*. Cambridge University Press, 1999.
- [22] Jürgen Bokowski. Oriented matroids. In Peter Gruber and Jörg Wills, editors, *Handbook of Convex Geometry*, volume A, B, pages 555–602. North Holland, 1993.

- [23] Jürgen Bokowski, Susanne Mock, and Ileana Streinu. On the Folkman-Lawrence topological representation theorem for oriented matroids in rank 3. *European J. of Combinatorics*, 22(5):601–615, 2001.
- [24] Ciprian Borcea and Ileana Streinu. The number of embeddings of minimally rigid graphs. *Discrete and Computational Geometry*, 31:287–303, 2004. A preliminary version appeared in Proc. ACM Symp. Comput. Geometry, Barcelona, 2002.
- [25] Hervé Brönnimann. The pseudo triangulation template toolkit. <http://geometry.poly.edu/pstoolkit/>, 2001.
- [26] Hervé Brönnimann, Lutz Kettner, Michel Pocchiola, and Jack Snoeyink. Counting and enumerating pseudo-triangulations with the greedy flip algorithm. In *Proc. 13th Canad. Conf. Comp. Geom.*, 2001.
- [27] John Canny. *Complexity of Robot Motion Planning*. MIT Press, Cambridge, MA, 1988.
- [28] Bernard Chazelle. Triangulating a simple polygon in linear time. *Discrete Comput. Geom.*, 6(5):485–524, 1991.
- [29] Roxana Cocan and Joseph O’Rourke. Polygonal chains cannot lock in 4d. *Discrete & Computational Geometry*, 20(3):105–129, November 2001.
- [30] Robert Connelley, Erik Demaine, and Günter Rote. Straightening polygonal arcs and convexifying polygonal cycles. In *Proc. 41st Annual Symp. on Found. of Computer Science (FOCS), Redondo Beach, California*, pages 432–452, 2000. Journal version in *Discrete and Computational Geometry*, 30(5), 205–239, 2003.
- [31] Robert Connelly, Erik D. Demaine, and Günter Rote. Infinitesimally locked self-touching linkages with applications to locked trees. In Jorge Calvo, Kenneth Millett, and Eric Rawdon, editors, *Physical Knots: Knotting, Linking, and Folding of Geometric Objects in R^3* , pages 287–311. American Mathematical Society, 2002. Collection of papers from the Special Session on Physical Knotting and Unknotting at the AMS Spring Western Section Meeting, Las Vegas, Nevada, April 21–22, 2001.
- [32] Henri Crapo. Two-tree-three decompositions. unpublished manuscript, 1990.
- [33] Henri Crapo and Walter Whiteley. Plane self-stresses and projected polyhedra I: the basic pattern. *Structural Topology*, 20:55–77, 1993.
- [34] Paul Erdős. Problem number 3763. *American Mathematical Monthly*, 46(627), 1935.
- [35] Israel Gel’fand, Mikhail Kapranov, and Andrei Zelevinsky. Newton polytopes of principal a-determinants. *Soviet Math. Doklady*, 40:278–281, 1990.
- [36] H. Gluck. Almost all simply connected closed surfaces are rigid. In *Geometric Topology*, number 438 in Lecture Notes in Mathematics, pages 225–239. Springer Verlag, Berlin, 1975.
- [37] Jacob E. Goodman and Richard Pollack. Allowable sequences and order types in discrete and computational geometry. In János Pach, editor, *New Trends in Discrete and Computational Geometry*. Springer Verlag, 1993.

- [38] Jack Graver, Brigitte Servatius, and Herman Servatius. *Combinatorial Rigidity*. Graduate Studies in Mathematics vol. 2. American Mathematical Society, 1993.
- [39] Leonidas Guibas and Jorje Stolfi. Primitives for the manipulation of general subdivisions and the computation of voronoi diagrams. *ACM Transactions on Graphics*, 4:74–123, 1985.
- [40] Leonidas J. Guibas, J. Hershberger, D. Leven, Micha Sharir, and R. E. Tarjan. Linear-time algorithms for visibility and shortest path problems inside triangulated simple polygons. *Algorithmica*, 2:209–233, 1987.
- [41] Ruth Haas, David Orden, Günter Rote, Francisco Santos, Brigitte Servatius, Herman Servatius, Diane Souvaine, Ileana Streinu, and Walter Whiteley. Planar minimally rigid graphs and pseudo-triangulations. In *ACM Symp. Computational Geometry (SoCG)*, pages 154–163, San Diego, California, June 2003.
- [42] L. Henneberg. *Die graphische Statik der starren Systeme*. Johnson Reprint 1968. Leipzig, 1911.
- [43] John Hopcroft, Deborah Joseph, and Sue Whitesides. On the movement of robot arms in 2-dimensional bounded regions. *SIAM Journal of Computing*, 14:315–333, 1985.
- [44] Deborah Joseph and W.H. Platinga. On the complexity of reachability and motion planning questions. In *Proceedings ACM Symposium on Computational Geometry (SoCG)*, pages 62–66, 1985.
- [45] Micha Kapovich and John Millson. On the moduli spaces of polygons in the euclidean plane. *Journal of Differential Geometry*, 42(1):133–164, 1995.
- [46] Lutz Kettner. Pseudo triangulation workbench. <http://www.cs.unc.edu/Research/compgeom/pseudoT/>, 2001.
- [47] Lutz Kettner, David Kirkpatrick, Andrea Mantler, Jack Snoeyink, Bettina Speckmann, and Fumihiko Takeuchi. Tight degree bounds for pseudo-triangulations of points. *Comp. Geom. Theory. Applications*, 25:1–12, 2003.
- [48] Rob Kirby. Problems in low-dimensional topology. <http://www.math.berkeley.edu/~kirby>, 1995.
- [49] David Kirkpatrick, Jack Snoeyink, and Bettina Speckmann. Kinetic collision detection for simple polygons. *International Journal of Computational Geometry and Applications*, 12(1 and 2):3–27, 2002.
- [50] David Kirkpatrick and Bettina Speckmann. Kinetic maintenance of context-sensitive hierarchical representations for disjoint simple polygons. In *Proc. 18th ACM Symposium on Computational Geometry (SoCG)*, pages 179–188, 2002.
- [51] G. Laman. On graphs and rigidity of plane skeletal structures. *J. Engrg. Math.*, 4:331–340, 1970.
- [52] William J. Lenhart and Sue Whitesides. Reconfiguring closed polygonal chains in euclidian d -space. *Discrete & Computational Geometry*, 13:123–140, 1995.

- [53] Laszlo Lovasz and Y. Yemini. On generic rigidity in the plane. *SIAM J. Algebraic and Discrete Methods*, 3(1):91–98, 1982.
- [54] Anna Lubiw and Sue Whitesides, editors. *Workshop on Folding and Unfolding*, Barbados, 1998. McGill University Bellairs Institute. Informal proceedings.
- [55] James Clerk Maxwell. On reciprocal figures and diagrams of forces. *Philosophical Magazine*, 27:250–261, 1864.
- [56] James Clerk Maxwell. On reciprocal figures, frameworks and diagrams of forces. *Transactions of the Royal Society Edinburgh*, 26:1–40, 1870.
- [57] James Clerk Maxwell. On bow’s method of drawing diagrams in graphical statics, with illustrations from peaucellier’s linkage. *Cambridge Phil. Soc. Proc.*, 2:407–414, 1876.
- [58] R. James Milgram and Jeff C. Trinkle. The geometry of configuration spaces for closed chains in two and three dimensions. *Homology Homotopy and Applications*, 2003. to appear.
- [59] David Orden, Guenter Rote, Francisco Santos, Brigitte Servatius, Herman Servatius, and Walter Whiteley. Non-crossing frameworks with non-crossing reciprocals. <http://arxiv.org/abs/math.MG/0309156>, 2003.
- [60] David Orden and Francisco Santos. The polytope of non-crossing graphs on a planar point set. <http://arxiv.org/abs/math.CO/0302126>, February 2003.
- [61] David Orden, Brigitte Servatius, Herman Servatius, and Francisco Santos. Combinatorial pseudo-triangulations. <http://arxiv.org/abs/math.CO/0307370>, July 2003.
- [62] Michel Pocchiola and Gert Vegter. Computing the visibility graph via pseudo-triangulations. In *Proceedings of the 11th annual Symposium on Computational Geometry (SoCG)*, pages 248–257. ACM Press, 1995.
- [63] Michel Pocchiola and Gert Vegter. Topologically sweeping visibility complexes via pseudo-triangulations. *Discrete & Computational Geometry*, 16(4):419–453, Dec. 1996.
- [64] F. P. Preparata and M. I. Shamos. *Computational Geometry: An Introduction*. Springer-Verlag, New York, NY, 1985.
- [65] Dana Randall, Günter Rote, Francisco Santos, and Jack Snoeyink. Counting triangulations and pseudo-triangulations of wheels. In *Proceedings 13th Canad. Conf. Comput. Geom.*, pages 149–152, 2001. <http://compgeo.math.uwaterloo.ca/cccg01/proceedings>, cite-seer.nj.nec.com/randall01counting.html.
- [66] András Recski. *Matroid Theory and Its Applications*. Springer Verlag, 1989.
- [67] Jürgen Richter-Gebert and Ulrich Kortenkamp. *Cinderella, an interactive software package for Geometry*. Springer Verlag, 1999.
- [68] Günter Rote, Francisco Santos, and Ileana Streinu. Expansive motions and the polytope of pointed pseudo-triangulations. In Janos Pach Boris Aronov, Saugata Basu and Micha Sharir, editors, *Discrete and Computational Geometry - The Goodman-Pollack Festschrift, Algorithms and Combinatorics*, pages 699–736. Springer Verlag, Berlin, 2003.

- [69] Günter Rote, Cao An Wang, Lusheng Wang, and Yinfeng Xu. On constrained minimum pseudo triangulations. In *Proceedings 9th Annual International Conference on Computing and Combinatorics, COCOON 2003, Big Sky, MT, USA, July 25-28, 2003*, volume 2697 of *Lecture Notes in Computer Science*, pages 445–454. Springer, 2003.
- [70] Jack Schwartz and Micha Sharir. On the piano mover’s problem II: general techniques for computing topological properties of real algebraic manifolds. *Adv. in Appl. Math.*, 4:298–351, 1983.
- [71] Bettina Speckmann and Csaba Tóth. Allocating vertex π -guards in simple polygons via pseudo-triangulations. In *Proc. ACM-SIAM Symp. Discrete Algorithms (SODA)*, pages 109–118, 2003. To appear in *Discrete and Computational Geometry*, 2004.
- [72] Ileana Streinu. Clusters of stars. In *Proceedings of the thirteenth annual Symposium on Computational Geometry*, pages 439–441. ACM Press, 1997.
- [73] Ileana Streinu. A combinatorial approach to planar non-colliding robot arm motion planning. In *ACM/IEEE Symposium on Foundations of Computer Science*, pages 443–453, 2000.
- [74] Ileana Streinu. Combinatorial pseudo-triangulations and partial oriented matroids. Draft, December 2003.
- [75] Ileana Streinu. Combinatorial roadmaps in configuration spaces of simple planar polygons. In Saugata Basu and Laureano Gonzalez-Vega, editors, *Proceedings of the DIMACS Workshop on on Algorithmic and Quantitative Aspects of Real Algebraic Geometry in Mathematics and Computer Science*, pages 181–206. DIMACS, 2003.
- [76] Ileana Streinu. Parallel redrawing pseudo triangulation mechanisms. Manuscript, 2003.
- [77] Ileana Streinu and Walter Whiteley. Single-vertex origami and 3-dimensional expansive motions. To appear, Proc. Japan Conf. Discrete and Comp. Geometry, Tokai Univ., Oct. 2004.
- [78] T.S. Tay and Walter Whiteley. Generating isostatic graphs. *Structural Topology*, 11:21–68, 1985.
- [79] Godfried Toussaint. The erdős-nagy theorem and its ramifications. In *Proceedings of the Canadian Conference on Computational Geometry*, 1999.
- [80] Jeff C. Trinkle and R. James Milgram. Complete path planning for closed kinematic chains with spherical joints. *International Journal of Robotics Research*, 21(9):773–789, Dec. 2002.
- [81] Walter Whiteley. Motions and stresses of projected polyhedra. *Structural Topology*, 7:13–38, 1982.
- [82] Walter Whiteley. Some matroids from discrete applied geometry. In J. Oxley J. Bonin and B. Servatius, editors, *Matroid Theory*, volume 197 of *Contemporary Mathematics*, pages 171–311. American Mathematical Society, 1996.

- [83] Walter Whiteley. Rigidity and scene analysis. In Jacob E. Goodman and Joseph O'Rourke, editors, *Handbook of Discrete and Computational Geometry*, pages 893–916. CRC Press, Boca Raton New York, first edition, 1997.
- [84] Sue Whitesides. Algorithmic issues in the geometry of planar linkage movement. *Australian Computer Journal, Special Issue on Algorithms*, 24(2):42–50, 1992.

Single-Vertex Origami and Spherical Expansive Motions

Ileana Streinu¹ * and Walter Whiteley² **

¹ Computer Science Department, Smith College, Northampton, MA 01063, USA.

streinu@cs.smith.edu, <http://cs.smith.edu/~streinu>

² Department of Mathematics, York University, Toronto, M3J 1P3, Canada.

whiteley@mathstat.yorku.ca, <http://mathstat.yorku.ca/~whiteley>

Abstract. We prove that all single-vertex origami shapes are reachable from the open flat state via simple, non-crossing motions. We also consider *conical* paper, where the total sum of the cone angles centered at the origami vertex is not 2π . For an angle sum less than 2π , the configuration space of origami shapes compatible with the given metric has two components, and within each component, a shape can always be reconfigured via simple (non-crossing) motions. Such a reconfiguration may not always be possible for an angle sum larger than 2π .

The proofs rely on natural extensions to the sphere of planar Euclidean rigidity results regarding the existence and combinatorial characterization of expansive motions. In particular, we extend the concept of a pseudo-triangulation from the Euclidean to the spherical case. As a consequence, we formulate a set of *necessary conditions* that must be satisfied by three-dimensional generalizations of pointed pseudo-triangulations.

1 Introduction

Imagine making creases in a flat sheet of paper, all of them originating at a single vertex, and then folding along the creases *without tearing, stretching or bending the paper*, to obtain a three dimensional origami shape. Assume that the planar regions bounded by creases behave more like metal sheets than paper, i.e. they move *rigidly*, and do not go through each other during the motion. See Fig. 1. The *single-vertex origami problem* asks: *are there origami shapes which are compatible with the creases and the induced metric of the paper, but which cannot be folded by such a process?*

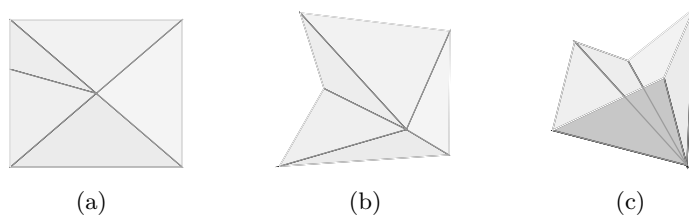


Fig. 1. A single-vertex origami fold: (a) the creased sheet of paper; (b, c) two of its possible folded shapes.

* Supported by NSF grants CCR-0105507 and NSF-DARPA CARGO-0310661.

** Supported by grants from NSERC (Canada) and NIH (US).

Here, we answer this question and generalize it in several ways. We prove that all single-vertex origami shapes can be folded from the flat state. This implies that the configuration space of all origami shapes with the same crease pattern is *connected*: any shape can be reconfigured into any other shape, via simple (non-self-intersecting) motions. We begin by formulating the problem in terms of conical *panel-and-hinge structures*, which have incident hinge-axes and in terms of *spherical polygonal linkages*. If the spherical perimeter (defined below) is no more than 2π , we show that then they can be reconfigured to a spherically convex configuration. But if the perimeter is more than 2π , then the configuration space may be disconnected. We leave open the question of whether simple spherical polygonal *paths* of length between π and 2π , with *short* links (less than π) is connected or not.

Our proofs rely on a generalization to the sphere of the planar *Carpenter’s Rule Problem*, which asks whether every simple planar polygon can be unfolded to a convex position, in such a way that the edges maintain their lengths and do not cross throughout the motion. An *expansive motion* never decreases any interdistance between two points, therefore no collisions may occur. The two solutions of the Carpenter’s Rule problem for simple planar polygonal linkages [4, 17] rely on expansive motions. In dimension two, the infinitesimal expansive motions are well understood. They form a polyhedral cone [15] whose extreme rays have a combinatorial characterization, given by pointed pseudo-triangulation mechanisms [17]. Their three-dimensional counterparts also form a polyhedral cone, defined by similar linear inequalities, but finding a combinatorial interpretation for its rays has so far remained elusive.

In this paper, we also initiate the study of expansive motions on the sphere and in 3d and give the first provable classes of spherical and 3d expansive mechanisms with one-degree-of-freedom: hemispherical pseudo-triangulations, resp. pointed *cone pseudo-triangulations*. By applying them to the spherical Carpenter’s Rule Problem of perimeter $\leq 2\pi$, we obtain the proof that all *simple* folds can be opened to a spherical convex position.

Historical background. Computational origami is a relatively recent endeavor, see [6] for a survey. The mathematical and computational origami literature addresses questions of feasibility, characterization and NP-hardness of flat-folds, as well as applications, see e.g. [13, 1, 7]. According to Tom Hull [10], only two published articles deal with the mathematics of rigid origami [9, 14].

The topology of the configuration space for spherical linkages and single-vertex origami folds (allowing self-crossings) is studied in [12, 11].

Polygonal linkages in the plane have received a lot of attention in recent years. Relevant for our paper are the previously mentioned results on the Carpenter’s Rule Problem using expansive motions [4] and pseudo-triangulations [17]. To the best of our knowledge, there are no other results studying or applying spherical or 3d expansive motions.

Rigidity for non-Euclidean geometries (including spherical and hyperbolic) is the topic of an unpublished paper of the second author [16]. The projective connections between motions of cones, motions on spheres, and motions in plane projections go back to [19]. We achieve the extension of the Carpenter’s Rule problem to spherical polygons and single-vertex origami folds via spherical geometry techniques, building on ideas from [19, 16] but handling signed motions (expansive and contractive).

2 Definitions and Preliminaries

For rigidity theoretic terminology and concepts, we refer the reader to the classical monograph [8] and the handbook chapter [20]. For pseudo-triangulations, see [17] and [15].

Frameworks in two and three dimensions. A bar-and-joint framework (simply, a framework) $G(p)$ is a graph $G = (V, E)$, $V = [n] := \{1, \dots, n\}$, embedded on a set of points $p = \{p_1, \dots, p_n\}$, $p_i \in \mathbb{R}^d$. In this paper $d = 2$ or $d = 3$. A pair of indices $(i, j) \in [n]^2$ may be denoted simply as ij . The embedded edges (segments) $p_i p_j$ are also called *bars*, and their endpoints p_i are called *joints*. When the underlying graph is a path or a cycle, the framework is called a *chain* (open, resp. closed).

Infinitesimal rigidity of frameworks. An *infinitesimal motion* of a framework $G(p)$ is a set of velocity vectors $v = \{v_1, \dots, v_n\}$, $v_i \in \mathbb{R}^d$ preserving the lengths of the bars:

$$\langle p_i - p_j, v_i - v_j \rangle = 0, \forall ij \in E$$

An infinitesimal motion is *trivial* if it is a rigid transformation of the whole space. A framework is *infinitesimally rigid* if it has only trivial infinitesimal motions, and *infinitesimally flexible* otherwise.

A *flex* or motion of a framework is a set of continuous point trajectories

$$p(t) = \{p_1(t), \dots, p_n(t)\}$$

which preserve the edge lengths $l_{ij}^2 = \langle p_i - p_j, p_i - p_j \rangle, \forall ij \in E$ of the initial framework $G(p) = G(p(0))$ at any moment in time t :

$$\langle p_i(t) - p_j(t), p_i(t) - p_j(t) \rangle = l_{ij}^2, \forall ij \in E$$

Expansion and contraction. Given a point set p and infinitesimal velocities v , let us denote by ε_{ij} the quantity $\varepsilon_{ij}(p, v) := \langle p_i - p_j, v_i - v_j \rangle$. For a pair ij of indices (not necessarily an edge of a graph G), we say that the diagonal ij *expands* if $\varepsilon_{ij} > 0$, *contracts* if $\varepsilon_{ij} < 0$ or is *frozen* if $\varepsilon_{ij} = 0$. An infinitesimal motion v of $G(p)$ is *expansive* (resp. *contractive*) if all the non-frozen diagonals expand (resp. contract). A framework is infinitesimally *expansive* if it is infinitesimally flexible and supports a non-trivial infinitesimally expansive motion.

Pointed pseudo-triangulations and mechanisms. A special class of planar frameworks are those with no crossing edges and where every vertex is *pointed*: incident to an angle larger than π . Such frameworks are planar graph embeddings and can have at most $2n - 3$ edges. When they have the maximum number of edges, the outer face is convex and all the internal faces are *pseudo-triangles*: simple polygons with exactly three inner convex angles. As frameworks, pointed pseudo-triangulations are *minimally rigid*. Removing any edge makes then *flexible* mechanisms, with one degree of freedom. If the removed edge is a convex hull edge, the mechanism is *expansive*: the unique motion that increases the length of the removed convex hull edge, never decreases any distance between a pair of points. See [17] and Fig. 2(a).

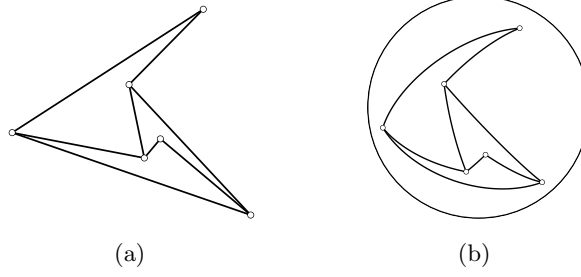


Fig. 2. A pointed pseudo-triangulation mechanism (a) in the plane, and (b) on the sphere.

Panel and hinge structures. A bounded *panel* is a simple polygon embedded flat in 3d, and intended to behave like a rigid object, i.e. to remain flat and to maintain its metric properties (edge lengths and angles). The simplest case is a triangle. We will also work with unbounded panels which are wedges defined by two semi-infinite rays.

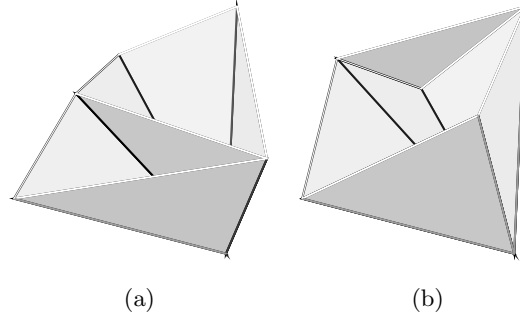


Fig. 3. Examples of panel-and-hinge structures. (a) Some hinges may be incident with several panels. (b) A triangulated polyhedral surface is a special case of a panel and hinge structure.

A *hinge* is a line segment or ray common to at least two panels. In this paper we consider only hinges that appear as complete bounding edges of their incident panels. In particular, a hinge does not run through the interior of a panel and does not extend beyond its boundary. The panels are attached rigidly to hinges: they are allowed to rotate about hinges, but not to slide along them.

A *panel-and-hinge* structure is a collection of panels connected via hinges. Examples include those in Fig. 1, where all hinges are concurrent, and Fig. 3, where each hinge is incident to two panels.

Conical panel-and-hinge structures. In this paper we work only with *conical panel-and-hinge structures*, in which all the hinges are concurrent to a unique vertex called the *cone vertex*, as in Fig. 3(a). In this case, each panel contains at most two hinges. The conical structures are *bounded*, if all panels are bounded polygons and the hinges are line segments, as in Fig. 3(a), or *unbounded*, when the hinges are infinite rays and the panels are wedges. For the purpose of this paper, in the bounded case it suffices to work only with triangular faces. We distinguish

two cases, *pointed* and *non-pointed conical structures*, depending on whether the cone vertex is pointed (all incident segments are contained in a half-space defined by a plane through the cone vertex) or not. See Fig. 4.

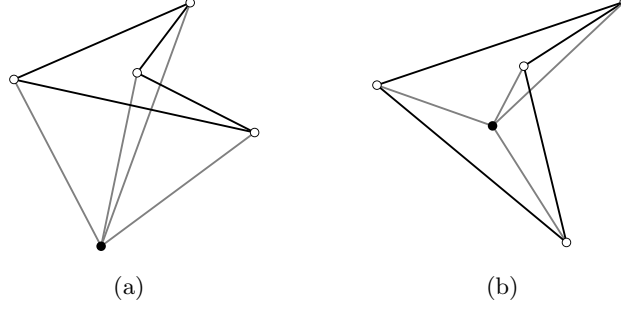


Fig. 4. (a) A pointed and (b) a non-pointed conical framework arising from conical panel-and-hinge structures. The cone vertex is black.

From conical panel-and-hinge structures to bar-and-joint frameworks. To each bounded conical panel-and-hinge structure we associate a bar-and-joint framework $G_0(p)$ in 3d. The vertices of G_0 correspond to those of the conical structure and the edges to sides of the triangular panels. The cone-vertex is always labeled with 0, and is incident with all the other vertices, labeled from 1 to n . G_0 is embedded as $G_0(p)$ on the set of points bounding the triangular panels, $p_i \in R^3$, $i = 0, 1, \dots, n$. Such a bar-and-joint framework is called a *cone framework*.

An *infinitesimal motion* of a bounded conical panel-and-hinge structure is an infinitesimal motion of the associated conical bar-and-joint framework.

For unbounded conical panel-and-hinge structures, we first bound the panels by *cutting* them into triangles. Then we define the infinitesimal motion of the bounded structure. The following straightforward lemma shows that the motion can be uniquely extended to all the points of the original unbounded panels.

Lemma 1. (Infinitesimal velocities of arbitrary points) *Any point p' situated on the supporting line of a hinge p_0p_i or on the supporting plane of a panel $p_0p_ip_j$ can be assigned an infinitesimal velocity v' uniquely determined by the velocities of the points bounding the hinge or panel.*

Spherical frameworks. If we intersect an unbounded conical structure with a sphere centered at the cone vertex, we obtain a *spherical framework*. It consists of *spherical bars* along great-circle arcs on the sphere, and joints which are the intersection of the hinges with the sphere. The incidence structure of a spherical framework is a graph G obtained from G_0 by deleting the cone-vertex 0 and its incident edges. The remaining graph G has the vertex set $[n]$. The *length* of a spherical edge ij is the size of the angle $\angle p_ip_0p_j$ centered at the cone vertex.

A spherical framework fully contained in a hemisphere is called *hemispherical*. It arises from a pointed conical panel-and-hinge structure. See Fig. 5.

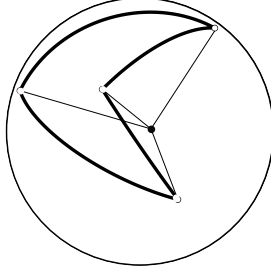


Fig. 5. A hemispherical framework, and the corresponding pointed conical panel-and-hinge structure. The cone-vertex (black) is the center of the sphere.

3 Infinitesimal rigidity of conical and spherical frameworks

In this section we show how to associate to every *planar framework* a 3d pointed *conical framework*, and therefore also a hemispherical framework and a pointed conical panel-and-hinge structure, and vice-versa. The association preserves infinitesimal rigidity and maintains the expansion pattern.

The cone of a planar framework. Let $G(p)$ be a planar framework embedded in R^2 , viewed as the affine plane $z = 1$ in R^3 . The *cone* over G is the graph $G_0 = (V_0, E_0)$, $V_0 = V \cup \{0\}$ (0 is the *cone vertex*) and $E_0 = E \cup \{0i\}_{i \in V}$. The *canonical conical framework* $G_0(p)$ over $G(p)$ with center $p_0 \in R^3$ is an embedding of the cone G_0 over G so that it extends $G(p)$ and embeds the cone vertex at the center p_0 . In this paper we take $p_0 = 0$. More generally, a cone framework $G_0(q)$ over $G(p)$ will have the cone vertex at q_0 and the i th vertex q_i on the line p_0p_i .

Infinitesimal motions of planar and conical frameworks. The association we just described induces a *projection map* carrying a conical framework³ $G_0(q)$ with no points on the plane $z = 0$ into a framework $G(p)$ in R^2 embedded in the plane $z = 1$ (and vice-versa). We now turn our attention to establishing and proving the connection between infinitesimal motions of a planar framework and the associated conical framework.

Let $G_0(q)$ be a conical framework with the cone vertex at the origin, $q = (q_0, q_1, \dots, q_n)$, $q_0 = 0$ and $q_i = (x_i, y_i, z_i)$, $z_i \neq 0$. Let $u = (u_0, u_1, \dots, u_n)$, $u_0 = 0$ be an infinitesimal motion of the framework $G_0(q)$ in R^3 , with the cone vertex pinned down. Let $\mathbf{e}$ the unit vector along the z -axis. See Fig. 6.

Proposition 1. *Consider the map $R^3 \mapsto R^2$ taking the points $q_i = (x_i, y_i, z_i)$ on the sphere to points $p_i = (\frac{x_i}{z_i}, \frac{y_i}{z_i}, 1)$ in the $z = 1$ plane and velocities u_i to:*

$$v_i = \frac{1}{z_i} (u_i - \langle u_i, \mathbf{e} \rangle \mathbf{e})$$

Then $u = (0, u_1, \dots, u_n)$ is an infinitesimal motion of the conical framework $G_0(q)$ in R^3 iff $v = (v_1, \dots, v_n)$ is an infinitesimal motion in R^2 of the framework $G(p)$.

³ In this section, we denote by p a planar point set and by q a 3d point set.

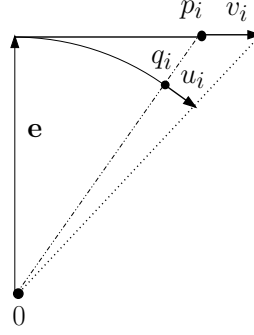


Fig. 6. The projection map, shown here in a section along the plane spanned by a point q_i on the sphere and the z -axis. It takes a point q_i on the sphere to a point p_i in the $z = 1$ plane, and an infinitesimal velocity u_i of q_i to an infinitesimal velocity v_i of p_i .

If two points q_i and q_j are in the same hemisphere ($z > 0$ or $z < 0$), then u is expansive for the pair q_i, q_j iff v is expansive on p_i, p_j in the plane. If q_i and q_j are on opposite hemispheres, then u is expansive for the pair q_i, q_j iff v is contractive on p_i, p_j in the plane.

Proof: The bars from the cone vertex give the equations $\langle q_i - q_0, u_i - u_0 \rangle = \langle q_i, u_i \rangle = 0$. Applying them on i and j , the expansion values become:

$$\begin{aligned} \langle p_i - p_j, v_i - v_j \rangle &= \left\langle \frac{q_i}{z_i} - \frac{q_j}{z_j}, \frac{1}{z_i} (u_i - \langle u_i, \mathbf{e} \rangle \mathbf{e}) - \frac{1}{z_j} (u_j - \langle u_j, \mathbf{e} \rangle \mathbf{e}) \right\rangle = \\ &= -\frac{\langle q_i, u_j \rangle + \langle q_j, u_i \rangle}{z_i z_j} + \left[\frac{\langle u_i, \mathbf{e} \rangle}{z_i} - \frac{\langle u_j, \mathbf{e} \rangle}{z_j} - \frac{\langle u_i, \mathbf{e} \rangle}{z_i} + \frac{\langle u_j, \mathbf{e} \rangle}{z_j} \right] = \\ &= -\frac{\langle q_i, u_j \rangle + \langle q_j, u_i \rangle}{\langle \mathbf{e}, q_i \rangle \langle \mathbf{e}, q_j \rangle} + \frac{1}{z_i z_j} [\langle q_i, u_i \rangle - \langle q_i, u_j \rangle - \langle q_j, u_i \rangle + \langle q_j, u_j \rangle] = \frac{1}{z_i z_j} \langle q_i - q_j, u_i - u_j \rangle \end{aligned}$$

Thus the sign (positive, negative or zero) of $\langle p_i - p_j, v_i - v_j \rangle$ is the same as the sign of $\langle q_i - q_j, u_i - u_j \rangle$ when the two points q_i and q_j are in the same hemisphere ($z_i z_j > 0$) and opposite when in different hemispheres ($z_i z_j < 0$). $\square$

As a simple corollary we obtain:

Lemma 2. Let $G(p)$ be a planar framework and let $G_0(p)$ be an associated 3-dimensional cone framework. Then:

1. If $G(p)$ is infinitesimally rigid, then so is $G_0(p)$.
2. If $G(p)$ is infinitesimally expansive, and the cone vertex is pointed, then $G_0(p)$ is also expansive.

If the planar framework $G(p)$ is a pseudo-triangulation mechanism, then the spherical framework associated to the cone framework $G_0(p)$ will be called a *hemispherical pointed pseudo-triangulation mechanism*. See Fig. 2(b). Interpreting Lemma 2 in this case leads to:

Lemma 3. A hemispherical pseudo-triangulation mechanism is expansive.

As another corollary we also obtain the following simple necessary condition for 3d expansive motions. Given a graph G and a vertex i , let V_i be the vertex i together with the set of neighbors of i . The i th *star* G_i of G is the subgraph induced on V_i .

Lemma 4. *Let $G(q)$ be a spatial flexible framework and let $G_i(q)$ be the conical framework induced on the star G_i . If a conical framework $G_i(q)$ is not infinitesimally expansive, then $G(q)$ is not infinitesimally expansive. In other words, the following two local pointedness conditions are necessary for infinitesimal expansiveness:*

1. *The cone vertex of each star $G_i(q)$ is pointed (in 3d), and*
2. *Each induced cone framework projects to a pointed and non-crossing 2d framework.*

Part 2 of the previous lemma relies on the existence of expansive motions for planar pointed graphs (which are subgraphs of pointed pseudo-triangulations), see [17, 15]. Lemma 4 gives a necessary condition for a 3d framework to be expansive, and thus precludes any rigidity-theoretic generalization of planar pointed pseudo-triangulation mechanisms to 3d.

4 Convexifying spherical polygonal linkages

A spherical framework is a graph G embedded on the surface of a sphere, with edges going along arcs of great circles. The *length* of an edge $q_i q_j$ is the angle between the two line segments joining the center of the sphere to the two endpoints of the edge. To simplify the presentation, we will assume that edges have lengths different from 0 and π , i.e the endpoints of every edge are distinct and not-antipodal (the results hold even without this assumption).

A great-circle cuts the sphere into two hemispheres. With respect to a hemisphere, the defining great-circle is called its *equator*, and the point where the normal to the equator plane crosses the hemisphere is called the *pole* of the hemisphere.

A hemispherical framework is one lying on a hemisphere. A *proper* framework has all its edge lengths at most π . Of special interest are spherical and hemispherical polygons and polygonal paths. The perimeter of a polygon and the length of a polygonal path is the total sum of its edge lengths. A hemispherical polygon is *convex* if the projection on the plane going through the pole and parallel to the plane of the equator is a convex polygon.

Theorem 1. *Every simple hemispherical polygon of perimeter $\leq 2\pi$ and every simple hemispherical polygonal path of length $\leq \pi$ can be unfolded to a convex polygon using expansive motions.*

Proof: Note that any polygon of perimeter at most 2π must lie inside a closed hemisphere. If it lies inside an open hemisphere, we proceed with the following case. We return below to the extreme case, when there are antipodal points.

Project the polygon in an open hemisphere to a plane tangent to the pole of the hemisphere. Find an expansive infinitesimal motion of the planar polygon, e.g. one induced by a pointed pseudo-triangulation mechanism as in [17], or an arbitrary one obtained by a linear program as in [4]. Lemmas 2 and 3 imply that the induced cone framework is also moving expansively.

Move the spherical mechanism until an alignment event occurs, as in [17]. Notice that the alignment of two incident edges in a spherical pseudo-triangulation mechanism happens

exactly when the projection on the $z = 1$ plane has the corresponding edges aligned (but beware, the projection does not move as a planar rigid framework). The combinatorial structure of the spherical and planar pseudo-triangulations is the same. Therefore, as long as the polygon stays inside a hemisphere, there is always an expanding motion given by a pseudo-triangulation mechanism, and the convexification algorithm of [17] applies identically.

There is a concern that the motion might leave an open hemisphere. This occurs when two antipodal points appear. With two antipodal points in a polygon, there must be two paths joining the points, each of length at least π . This implies that the polygon must have length exactly 2π , and the two paths must each be great circle segments joining the antipodal points. In this situation, we may have the entire polygon on a single great circle, and we are finished with a flat spherical polygon. The alternative is that we have a ‘wedge’ formed by two intersecting great circles. Freezing one, we can rotate the other to make them flat, on opposite segments of a great circle. This motion is obviously expansive. Finally, we note that if we did not start with antipodal points, we can only achieve this configuration at an alignment event, so our previous process is complete and only requires a check for a wedge after each alignment event.

Finally, we note that if the algorithm terminates with a convex polygon which is not flat, then we have a *convex cone* from the center of the sphere which has a total angle of less than 2π . This means the spherical polygon had perimeter less than 2π . Therefore, any polygon of perimeter 2π will terminate with a flat configuration on a single great circle.

The polygonal path of length not more than 2π can be reduced to the closed polygon case by joining the endpoints of the path via a geodesic spherical path (which stays inside the hemisphere and bends at vertices and edges of the polygonal path). The total length of the added geodesic edges is at most the length of the polygon, so we have reduced to the previous case. Moreover, for any path less than π we could choose the additional edges to make the total length exactly 2π and terminate with a flat (collinear) polygonal path. If the length is exactly π , a limiting argument guarantees we can also achieve flatness. $\square$

Let’s emphasize as separate corollaries two ideas from the previous proof.

Lemma 5. *A spherical polygon of perimeter 2π can be unfolded to a great circle using expansive motions.*

Corollary 1. *If the geodesic line added between the endpoints of a spherical polygonal path doesn’t increase the total length to more than 2π , then the polygonal path can be made flat on a great circle using expansive motions.*

The case not covered by the previous corollary, the hemispherical polygonal path (even when its total length is under 2π) may need contractive (or partially contractive) motions to convexify. We conjecture that the convexification is always possible.

Conjecture 1. Every spherical polygonal path of length at most 2π can be flattened without collisions onto a great circle.

Just as in the planar case, we obtain:

Corollary 2. *Two similarly oriented configurations of a simple spherical polygon with fixed edge lengths and perimeter $\leq 2\pi$ lie in the same component of the configuration space.*

Indeed, one can move each one into convex position and then reverse one unfolding path to achieve the reconfiguring motion.

What happens for spherical polygons and paths longer than 2π ? They obviously cannot be convexified, since they do not fit on a great circle. The question is whether they can always be reconfigured to any other position. The following simple example shows that this may not be always possible.

Example 1. Consider a spherical quadrilateral with all edge lengths equal to $\frac{2\pi}{3} - \epsilon_i$, for four distinct values ϵ_i . Its perimeter is $\frac{8\pi}{3} - \sum \epsilon_i > 2\pi$, and any consecutive pair of edges adds to over π in length. The framework fits into a hemisphere in two ways, which cannot be reconfigured one to another without self-intersections.

Moreover, even when a reconfiguration of such spherical linkages is possible, a combination of expansive and contractive motions may have to be used.

5 Unfolding single-vertex origami

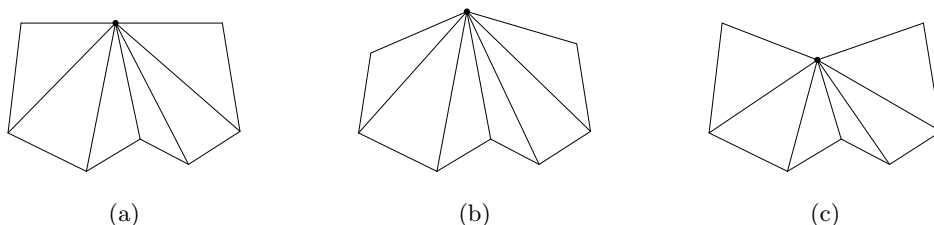


Fig. 7. Single-vertex origami with the fold-vertex on the boundary of the piece of paper: (a) on an edge, (b) at a convex corner and (c) at a reflex corner.

A single-vertex origami is a creased piece of paper with all the creases incident to one vertex. Assume first that the vertex lies in the interior of the piece of paper. For simplicity, assume that the paper has a polygonal boundary with one vertex on each crease line. The total angular length of the corresponding spherical polygon or path framework is 2π in this case. We also consider the case when the polygonal piece of paper has the fold-vertex situated on a boundary edge or at a corner of the paper. The total angular length of the corresponding spherical polygon or path framework is π in the first case, and it can be either strictly less than or larger than 2π in the second case, depending on the corner angle being convex or reflex. See Fig. 7. Theorem 1 implies:

Corollary 3. *Every simple single-vertex origami fold with the fold-vertex interior to the paper or interior to a boundary edge or situated at a convex vertex can be unfolded with expansive motions.*

Corollary 4. *The configuration space of simple single-vertex origamis with the fold-vertex interior to the paper, or to a boundary edge or situated at a convex vertex is connected. Two shapes can be reconfigured one into the other with simple, non-self-intersecting motions.*

Conjecture 1 extends to single-vertex origamis with the fold-vertex situated on a reflex corner of the paper.

References

1. M. Bern and B. Hayes. The complexity of flat origami. In *Proc. 7th ACM-SIAM Symp. on Discrete Algorithms (SODA)*, pages 175–183, Atlanta, January 1996.
2. 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 three dimensions. *Discrete & Computational Geometry*, 26(3):269–281, 2001.
3. J. Cantarella and H. Johnston. Nontrivial embeddings of polygonal intervals and unknots in 3-space. *J. Knot Theory Ramifications*, 7(8):1027–1039, 1998.
4. R. Connelly, E. Demaine and G. Rote. Straightening polygonal arcs and convexifying polygonal cycles. *Discrete & Computational Geometry*, 30(5):205–239, 2003.
5. H. Crapo and W. Whiteley. Statics of frameworks and motions of panel structures: a projective geometric introduction. *Structural Topology*, 6:43–82, 1982.
6. E. Demaine and M. Demaine. Recent results in computational origami. In *Proc. 3rd Int. Meeting of Origami Science, Math, and Education (OSME 2001)*, pages 3–16, Monterey, California, March 9–11 2001.
7. E. Demaine, M. Demaine, and J. Mitchell. Folding flat silhouettes and wrapping polyhedral packages: New results in computational origami. In *Proc. 15th ACM Symp. on Computational Geometry (SoCG’99)*, pages 105–114, Miami Beach, Florida, June 13–16 1999.
8. J. Graver, B. Servatius and H. Servatius. *Combinatorial Rigidity*. Graduate Studies in Mathematics vol. 2. American Mathematical Society, 1993.
9. D. Huffman. Curvature and creases: a primer on paper. *IEEE Transactions on Computers*, C-25(10):1010–1019, Oct. 1976.
10. T. Hull. Rigid origami. <http://www.merrimack.edu/~thull/rigid/rigid.html>, 2003.
11. M. Kapovich and J. Millson. Hodge theory and the art of paper-folding. *Publications of RIMS, Kyoto*, 33(1):1–33, 1997.
12. M. Kapovich and J. Millson. On the moduli space of a spherical polygonal linkage. *Canad. Math. Bull.*, 42(3):307–320, 1999.
13. R. Lang. A computational algorithm for origami design. In *Proc. 12th ACM Symposium on Computational Geometry*, pages 98–105, 1996.
14. K. Miura. A note on intrinsic geometry of origami. *Proc. First International Meeting of Origami Science and Technology (H. Huzita ed.)*, pages 239–249, 1989.
15. G. Rote, F. Santos and I. Streinu. Expansive motions and the polytope of pointed pseudo-triangulations. In J. Pach, B. Aronov, S. Basu and M. Sharir, editors, *Discrete and Computational Geometry - The Goodman-Pollack Festschrift*, Algorithms and Combinatorics, pages 699–736. Springer Verlag, Berlin, 2003.
16. F. Saliola and W. Whiteley. Notes on the equivalence of first-order rigidity in various geometries. Preprint, 2002.
17. I. Streinu. A combinatorial approach to planar non-colliding robot arm motion planning. In *Proc. ACM-IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 443–453, 2000.
18. T.S. Tay and W. Whiteley. Generating isostatic graphs. *Structural Topology*, 11:21–68, 1985.
19. W. Whiteley. Cones, infinity and one-story buildings. *Structural Topology*, 8:53–70, 1983.
20. W. Whiteley. Rigidity and scene analysis. In J.E. Goodman and J. O’Rourke, editors, *Handbook of Discrete and Computational Geometry*, pages 893–916. CRC Press, Boca Raton New York, first edition, 1997.

A Combinatorial Approach to Planar Non-colliding Robot Arm Motion Planning

Ileana Streinu *

Abstract

We propose a combinatorial approach to plan non-colliding motions for a planar robot arm. The approach works even with certain types of movable polygonal obstacles and flexible polygonal fences. This yields a very efficient deterministic algorithm for a category of robot arm motion planning problems with many degrees of freedom, for which the known general roadmap techniques have exponential complexity.

The main result is an efficient algorithm for convexifying a simple (open or closed) polygonal path with rigid non-intersecting motions in the plane. It works by computing in $\mathcal{O}(n^2)$ time a monotone mechanism with one degree of freedom, whose motion is controlled by the rotation of a single edge around one of its endpoints. As it moves, all the interdistances between pairs of points not joined by a bar are non-decreasing, thus guaranteeing non-collision. At most $\mathcal{O}(n^2)$ such motions suffice to reach a convex configuration of the original linkage. At each step, recomputing the next motion from the previous one can be done by local changes in at most $\mathcal{O}(n)$ time. Combinatorially, the structure of the mechanism is a pseudo-triangulation with a convex hull edge removed, which is shown to have several simple equivalent characterizations and to possess the right rigidity theoretic properties.

This settles algorithmically an open problem posed by W. Lenhart and Sue Whitesides, and independently by J. Mitchell, for which an existential answer has been recently found by Connelly, Demaine and Rote.

*Department of Computer Science, Smith College, Northampton, MA 01063, USA.
streinu@cs.smith.edu. Supported by NSF grant CCR-9731804.

1 Introduction

In this paper we present a new approach to a category of planar motion planning problems, including non-colliding unfoldings of open and closed chains (robot arms), even in the presence of certain types of obstacles (movable convex polygons and flexible fences). All of our constructions are elementary and are based on a novel class of planar embedded rigid graphs called *pseudo-triangulations* which possess rich combinatorial properties. We use techniques from Rigidity Theory and Visibility graphs, together with insights gained from Oriented Matroid Theory. The main ideas of our approach are best illustrated by the basic problem of moving a simple planar linkage or polygon to any other planar configuration, while remaining in the plane and without creating self-intersections along the way, for which we will describe a simple and efficient algorithm. This is done by first finding motions that convexify both configurations with monotone motions (defined below), then taking one path in reverse.

Frameworks and Robot arms. A bar-and-joint framework is a combinatorial graph $G = (V, E)$ embedded in the plane with rigid bars (fixed length straight line segments) corresponding to the edges. Edges can move freely in the plane around adjacent joints (vertices). The motions preserve rigidly the lengths of the graph edges, but impose no restriction on the non-edges, which may increase or decrease freely. In general, edges may cross during the motion, but in this paper we are interested in avoiding collisions and will work only with planar embedded graphs.

Of particular interest are the *monotone motions*, where all the pairwise interdistances between vertices never decrease during the motion, thus guaranteeing non-collision.

A *linkage* or *robot arm* is a planar framework whose underlying graph is a non-crossing path with n vertices, and a *closed chain* is a planar polygon on n vertices. *Straightening a linkage* means moving it continuously until all its vertices lie on one line with non-overlapping edges. *Convexifying a chain* means moving it to a position where it forms a simple convex polygon. Other types of frameworks of interest in

this paper include semi-simple polygons and pseudo-triangulations, defined below.

The Problem. W. Lenhart and S. Whitesides (1991), and J. Mitchell (1992) asked the following question: **Is it always possible to straighten a planar linkage, or to convexify a planar chain?** Recently, Connelly, Demaine and Rote [CDR] have answered this question in the affirmative. Their solution still leaves open the question: **Find, algorithmically, a sequence of simple motions to straighten a linkage, or to convexify a polygon.**

Previous Results on Reconfiguring Linkages. The general techniques for solving motion planning problems based on roadmaps work well on problems with bounded degrees of freedom, but yield exponential algorithms for high degree of freedom problems such as the one we are interested in. See [SS], [Ca88], [BPRa] and [BPRb]. In practical applications, probabilistic roadmap algorithms are used instead, see [Ka], [HA]. Under various conditions, problems about reconfiguration of linkages range in complexity from polynomial ([LW]) to NP- and even PSPACE-hard, see [HJW], [W92], [JP].

The particular problem of straightening bar-and-joint linkages and convexifying polygons has accumulated a distinguished history, with some approaches going back to a question of Erdős [Er]. See [To99] for a fascinating account. There are abundant connections with work done in the computational biology, chemistry and physics literature and motivated by topics such as protein folding or molecular modeling. When crossings are allowed, Lenhart and Whitesides [LW] have shown that the configuration space has at most two connected components and gave a linear algorithm for convexification based on simple motions (moving only a constant number of joints at a time). Recent results in the mathematics literature aim at understanding the topology of the configuration space of closed chains, but they allow crossings, see ([KM95]). The universality result of [KM98] is also for the general case of self-intersecting rigid frameworks. Following the McGill University Workshop on Folding in Barbados, organized by Anna Lubiw and Sue Whitesides in 1998, several results on reconfiguring with non-crossing motions have appeared: planar linkages using 3d motions ([BD+99], [AGP]), trees, 3 and higher-dimensional linkages ([BD+98], [COR]). The decisive result came in the beginning of this year, when Connelly et al. [CDR] announced the positive answer to the planar convexification problem: all chains can be convexified, all linkages can be straightened. Their approach is to first prove (using linear programming duality and Maxwell’s theorem, see [CW]) that there always exists a local, infinitesimal monotone motion, which can then be found using linear programming. Then they provide a global argument, showing the existence of a continuous deformation obtained by integrating the resulting vector field. To use this approach for finding a convexifying motion one would have to accept an approximate solution.

Our Results. We strengthen and provide an algorithmic proof for the result of [CDR]. Using the

motion planning terminology [CDR] have shown that the configuration space of planar non-crossing chains with the same orientation is connected. We show how to (algorithmically) find a path between any two such configurations. Our result extends to several other classes of embedded frameworks, including semi-simple polygons (the degenerate case of the convexification problem, not dealt with in any of the previous work). Along the way, we characterize a family of planar rigid frameworks called *pseudo-triangulations*, which yield 1DOF monotone mechanisms when a convex hull edge is removed, a result of independent interest in Rigidity Theory. The rich combinatorial and rigidity-theoretic properties of pseudo-triangulations which we present in this paper are likely to add to the applicability of this versatile data structure, a slight generalization of the one introduced by Pocchiola and Vegter [PV] in their study of the visibility complex and recently applied to kinetic geometric algorithms ([AB00], [BE99]).

Novelty. To the best of our knowledge, there are very few results on the motion planning of rigid frameworks when self-intersections are not allowed, and ours comes within a clean combinatorial setting. The rigidity theoretic properties of our generalized pseudo triangulations are novel, as is the pseudo-triangulation based approach to 1DOF monotone mechanisms for finding a path in configuration space. Along the way, we also give a generalization of a key lemma in [CDR] by showing the non-existence of self-stress in bar-and-cable tensegrity frameworks of a more complex structure than just simple polygons.

Proof Techniques and Overview of the Convexification Algorithm. The convexifying path, seen in the $(2n - 3)$ -dimensional configuration space (translations and rotations of the original chain being ruled out), is a sequence of at most $\mathcal{O}(n^2)$ curve segments (arcs), connecting continuously at their endpoints.

Each arc corresponds to the increasing motion induced by a monotone, one-degree-of-freedom (1DOF) mechanism induced by a planar pseudo-triangulation with a convex hull edge removed. The mechanism is constructed algorithmically by adding $n - 4$ bars to the original polygon. This mechanism is set in motion by pinning down an edge and rotating another edge around one of its joints. We show that this induces the whole framework to move *monotonically*, i.e. with non-decreasing interdistances between all pairs of vertices. One step of the convexification algorithm consists in moving this mechanism until it stops being a pseudo-triangulation. At that point, two of the edges of the framework become collinear. We either freeze a joint (if two edges of the polygon become collinear) and compute a new pseudo-triangulation for a polygon with one less vertex, or otherwise perform in constant time a local flip of the added diagonals. The correctness of this step requires particular care, as degenerate situations lead to all sorts of special situations in rigidity theory, which we are able to overcome in our case. Initially, a pseudo-triangulation is constructed incrementally in $\mathcal{O}(n^2)$ time. Each step eliminates at least one free diagonal from consider-

ation: it will never again become part of a future pseudo-triangulation. Hence the algorithm stops in at most $\mathcal{O}(n^2)$ such steps.

Pseudo-triangulations. A 1DOF monotone mechanism obtained from a pseudo-triangulation is a canonical representation of one of the many basic solutions, inducing monotone infinitesimal motions, that the linear programming approach of [CDR] would find for a certain position of the polygon in its configuration space.

We characterize pseudo-triangulations in at least four equivalent ways, exhibiting their rich combinatorial properties. Some of these are specialized versions of Laman’s $2n - 3$ count and Henneberg constructions from combinatorial rigidity (see [W96] or [GSS]). The proof of correctness of our approach derives from these properties, as well as from a generalization, from simple polygons to the wider class of pseudo-triangulation frameworks, of the approach used in [CDR] based on LP duality and Maxwell’s theorem [CW].

2 Definitions and Preliminaries

References. For rigidity theory terminology and basic results, we refer the reader to [Re], [W96], [W97] and [GSS]. In particular, rigidity, first-order and generic rigidity, as well as classical results on 2-dimensional rigidity such as Laman’s theorem, the Henneberg constructions, Lovasz and Yemini covering with two trees [LY] and Maxwell’s Theorem are to be found there. For oriented matroids, see [Bj93], although we won’t need more than the circular sequences of [GP] (see also [St]).

Notation and abbreviations. Our setting is the Euclidian plane. All index arithmetic is done mod n in the set $[n] := \{1, \dots, n\}$. We abbreviate “counter-clockwise” as *ccw* and “one-degree-of-freedom mechanism” as *1DOF mechanism*.

Acyclic set of vectors. A set of vectors in R^2 (with a common origin) is *acyclic* if it is strictly contained in a half-plane, and (*totally*) *cyclic* otherwise. The terminology comes from oriented matroid theory (see [Bj93]), an approach we won’t explain in this abstract, but which provided the guiding line in our search for combinatorial properties of rigid frameworks. For a set of acyclic vectors, there is no linear combination with strictly positive coefficients that sums the vectors to zero, while in the totally cyclic case there is always one. An intermediate case, treated as part of the totally cyclic case and reached on the boundary cases of our algorithm, is when two vectors are collinear and in opposite directions, with the extreme version of all the edges being collinear and at least two in opposite directions. Several vectors collinear but in the same direction, and the rest in acyclic relation with them belong to the acyclic case. The understanding of the boundary cases is crucial for the correctness of our approach, hence these detailed definitions. See the examples in Fig. 1. An *acyclic set of segments* has the property that all the edge vectors around each vertex are acyclic.

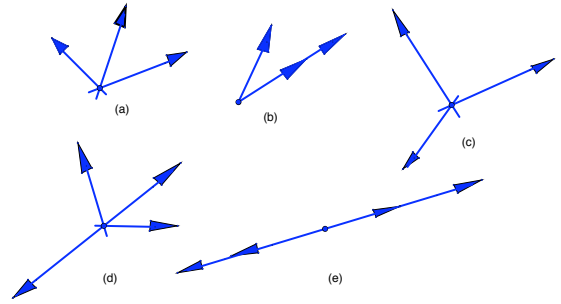


Figure 1: (a) and (b) Acyclic, (c) totally cyclic, (d) partially collinear and (e) fully collinear sets of vectors.

A **polygon on the ordered set of points** $\mathcal{P} = \{p_0, \dots, p_{n-1}\}$ is obtained by joining pairs of consecutive points with line segments (edges of the polygon). It is *simple*, if the edges do not cross or overlap. In this case, there is a well-defined interior and exterior of the polygon. We will assume that the vertices are labeled in such a way that the interior lies to the left, in a ccw traversal of the boundary of the polygon. More generally, we will allow repetitions in the set of points and overlaps of edges, and define a *semi-simple polygon* to be one with no proper crossings of edges, but with possible overlaps of edges and vertices. A proper crossing occurs when there exists a unique point of intersection of the two edges, which is strictly in their interior. Due to lengthy technicalities which may obscure the main ideas of our approach, we defer to the full paper the complete definition of all the special cases covered by our concept of semi-simple polygon. See some examples in Fig. 2. The motivation for introducing semi-simplicity (for polygons and later for frameworks and pseudo-triangulations) comes from the fact that it reflects what happens on the boundary of configuration spaces. Without proper care, the special boundary cases may lead to flawed arguments.

Frameworks. A *framework* is a graph $G = (V, E)$, $|V| = n$, embedded on a set of points $\mathcal{P} = \{p_1, \dots, p_n\}$. A *combinatorial framework* $F = (V, E, M)$ is a generalization of it, obtained by retaining (in M) only some combinatorial information about the underlying oriented matroid of the set of points $\mathcal{P}$. Since in this paper we work only with planar and acyclic frameworks, we do not give here the general definition. In our particular case, the information M retained from the embedding will be the signed circular sequence in which a *directed line* rotating ccw encounters the edge vectors (also including in certain situations all the visibility vectors) as they appear in ccw order around each vertex. An edge vector is recorded positively or negatively depending on whether the rotating line encounters it in its direction or in the opposite one. This concept is a specialization of the *local sequences*

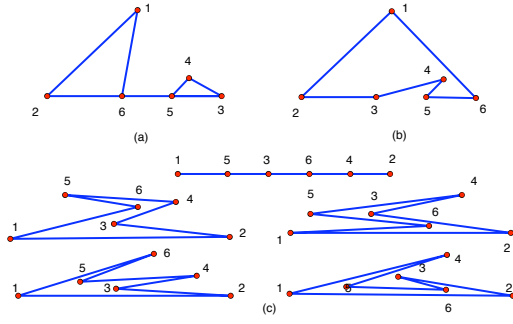


Figure 2: (a) A semi-simple and (b) a simple polygon on the same set of points, and (c) a semi-simple degenerate polygon and several possible interpretations (and perturbations) of it.

of Goodman and Pollack [GP] and retains (partial) oriented matroid structure from the underlying set of points P .

In particular, we can read off from this information the planar embedding of the framework and the cyclic or acyclic nature of the edge vectors at each vertex. A combinatorial framework is planar if its edges do not cross (except possibly at endpoints) and is acyclic if all the edge vectors around each vertex are acyclic. Special situations (not fully defined in this abstract) occur when we allow the faces of the planar embedding to be semi-simple polygons, or the edge vectors around a vertex to be partially or fully collinear.

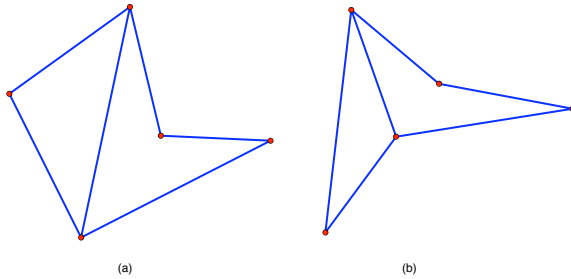


Figure 3: The same graph embedded as two combinatorially distinct frameworks, which are 1DOF mechanisms. (a) is monotone, (b) is not.

Generic rigidity, infinitesimal rigidity. A framework (planar or not) is (locally) *rigid* if its vertices cannot be moved while preserving the lengths of the edges, except for translations and rotations. Otherwise it is *flexible*. It is *infinitesimally flexible* if there

exists an assignment of velocity vectors v_i to each vertex p_i so that $\langle p_i - p_j, v_i - v_j \rangle = 0$, where $\langle, \rangle$ is the dot product. Otherwise it is infinitesimally (or first-order) rigid. A graph is *generically rigid* if it is rigid for all embedding on *generic* sets of points, see the rigidity theory references for the precise definition. Certain embedding of generically rigid graphs may be infinitesimally flexible, or even flexible. Laman's theorem provides a combinatorial characterization of generically rigid graphs: a graph on n vertices with exactly $2n - 3$ edges, and such that every subset of k vertices spans no more than $2k - 3$ edges. Henneberg constructions provide an inductive construction of generically rigid graphs via two types of local additions: type 1 is adding a vertex, joined by two new two adjacent edges to two previously constructed vertices, and type 2 is adding a new vertex, three adjacent edges to three old vertices such that at least two are joined by an edge, and dropping one of the existing edges among these three old vertices. The reader unfamiliar with these constructions may recognize, at least in the simplest applications of the type 1 steps, basic ways of producing triangulations. See Fig. 4.

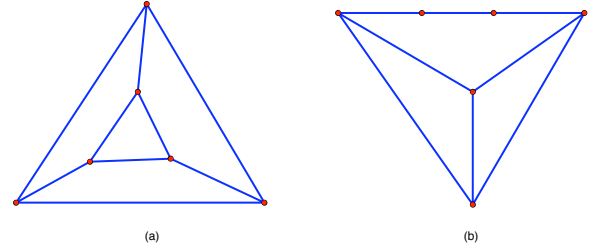


Figure 4: (a) A generically rigid graph in an infinitesimally flexible embedding. (b) A rigid framework with a non-generically rigid underlying graph.

One-degree-of-freedom mechanisms. A *mechanism* is an embedded framework which is flexible. Its degree of freedom (DOF) is the dimension of its configuration space (translations and rotations ruled out). A *generic minimally flexible graph* is a generically rigid graph with one edge removed (in particular, it has exactly $2n - 4$ edges). In a generic embedding, a minimally flexible graph is a *one degree of freedom (1DOF) mechanism*, but in other embeddings it may even be rigid, see Fig. 5.

Monotone mechanisms. A 1DOF mechanism is monotone if the non-rigid pairwise interdistances either all increase or all decrease during the local motion. See Fig. 3 for examples.

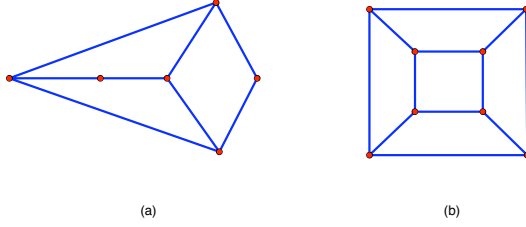


Figure 5: (a) A 1DOF mechanism (Peaucellier's linkage). (b) A framework with an underlying generically minimally flexible graph, but with a rigid (but not infinitesimally rigid) embedding.

3 Pseudo-Triangulations

A *pseudo-triangle* is a simple polygon with three vertices on its convex hull, joined by three inward convex polygonal chains. In particular, a triangle is a pseudo-triangle. A *pseudo-triangulation* of a set of points $\mathcal{P} = \{p_1, \dots, p_n\}$ is a set of edges (segments) tiling the convex hull of $\mathcal{P}$ into pseudo triangles, and which has the minimum number of edges. We will think of a pseudo-triangulation as a special kind of embedded framework. See Fig. 6.

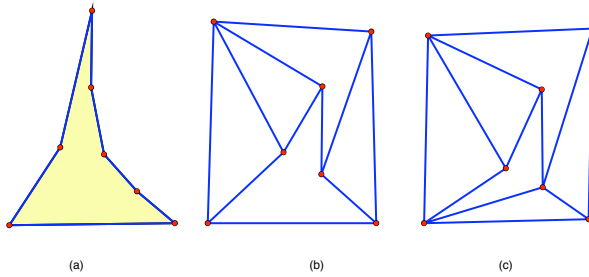


Figure 6: (a) A pseudo-triangle. (b) A pseudo-triangulation. (c) A partitioning into pseudo-triangles which is not a pseudo-triangulation.

Theorem 3.1 (Characterization of pseudo triangulations)

Let $G = (V, E)$ be a planar graph embedded on the set $\mathcal{P} = \{p_1, \dots, p_n\}$ of points. The following properties are equivalent.

1. G is a pseudo-triangulation.

2. The edges E of G form an acyclic set of segments, and E is maximal with this property (of being both planar and acyclic).
3. The faces of G are pseudo-triangles, the number of edges is $2n - 3$ and any set of k vertices has at most $2k - 3$ edges.
4. The set of edges E is acyclic, has $2n - 3$ elements and any subset of k vertices has at most $2k - 3$ edges.
5. G has a planar acyclic Henneberg construction, via two types of steps (to be described in the full paper).

Theorem 3.2 (Algorithms) Any acyclic set of edges can be extended to a pseudo-triangulation by randomly adding edges while preserving acyclicity. A canonical extension can be constructed deterministically in $O(n^2)$ time. If the set of points contains repetitions or collinearities, the resulting faces may be semi-simple pseudo triangles.

Proof. The proof for polygons has more structure, so we sketch here only this case. The pseudo-triangulation is constructed incrementally, adding one edge of the polygon at a time, in ccw order, starting at a vertex on the convex hull. This insures that the last step will reuse a previously inserted edge and won't necessitate any additional deletions. At each step of the insertion we add a vertex, a polygon edge and one additional edge. However, we might have to displace or modify several other edges, depending on whether the acyclicity condition, the planarity condition, or both are violated by the insertion of the new polygon edge. We show how to modify some of the added pseudo-triangulation edges using an argument which we call the *rubber band argument with snapping*, which locally modifies (in linear time) the edges adjacent to the new polygon edge to preserve acyclicity and planarity. The details are deferred to the full paper.

The pseudo-triangulation is not unique, in general. We have found several algorithms for performing the extension, each producing a certain type of canonical pseudo-triangulation with inductive or separation properties that may be of independent interest and may help in further speeding up the construction (we think that the $O(n^2)$ time method can be improved to $O(n \log n)$). In this paper we have not aimed at proposing the best algorithmic solutions for constructing pseudo-triangulations and concentrated instead on their properties and relationship to the convexification problem.

Theorem 3.3 (Rigidity property of pseudo triangulations) Pseudo-triangulations are infinitesimally rigid (and hence rigid).

The generic case follows from Laman's theorem and the properties of pseudo triangulations. The special cases need a careful analysis, which is deferred to the full paper.

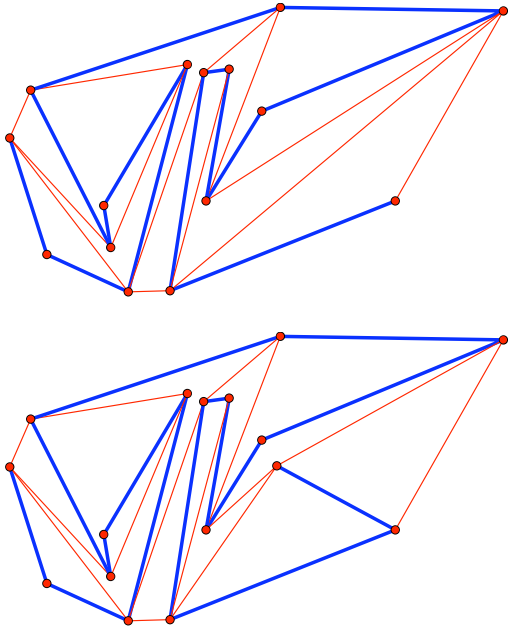


Figure 7: A typical step in the algorithm for computing the pseudo-triangulation of a polygon.

4 Monotone 1DOF mechanisms from pseudo triangulations

Theorem 4.1 (Main property of acyclic planar pseudo-triangulations) *A rigid bar-and-joint framework whose underlying graph is obtained by removing a convex hull edge from a pseudo-triangulation is a 1DOF monotone mechanism.*

Proof (sketch). The proof is an extension of the one in [CDR], using the acyclic property of the edges, and the linear independence in the configuration space of the $2n-4$ bars. In particular, we have strengthened their result, by showing that any framework which is generically independent (obtained from a generically rigid (dependent) framework by removing some bars), acyclic and does not contain all the convex hull edges is flexible and monotone.

We first show that there exists a *unique* infinitesimal motion (this is a stronger statement than the one in [CDR] for polygons, because we restricted the number of degrees of freedom of the mechanism). The proof depends only on the combinatorial type of the pseudo-triangulation, hence as long as the combinatorial substructure does not change, the motion continues. This step can be made more precise by an argument as in [CDR] (based on integration of the resulting vector field), and produces the desired trajectory in configuration space.

To prove the existence of the infinitesimal motion, we also just have to adapt the argument in [CDR],

using LP duality, Maxwell’s Theorem lifting and the mountain-valley ridge argument. It is based on the fact that cutting the polyhedral surface with a horizontal plane epsilon below the vertex with maximum z -coordinate in the lifting, one obtains a polygon, which will have at least three vertices on the convex hull that must correspond to mountain edges. The projection of the mountain edges on the plane of the section induces three cyclic vectors, which contradicts the acyclicity of the pseudo-triangulation. See Fig. 8, where M is the vertex where the maximum is attained. Just like in [CDR], the argument can be easily extended in the case when the maximum is attained on an edge or face, and acyclicity guarantees that we obtain a contradiction.

However, we need to add a few more details to the basic structure of [CDR] proof, because otherwise it will be incomplete in this general case. First, we need the assumption that the rank of the original system of equalities (corresponding to the bars of the framework) is $2n - 4$, otherwise there will be no basic solutions to the linear program. The argument based on Maxwell lifting is then used to show the existence of a positive solution, however we need extra care, since removing an edge from the framework to get the rank down to the critical value of $2n - 4$ does not, in itself, imply that there exists a strictly increasing motion on all diagonals despite the fact that the mountain-valley argument holds. Indeed, there are both monotone and non-monotone mechanisms arising from pseudo-triangulations by removing an edge, but only those obtained by removing a convex hull edge are monotone.

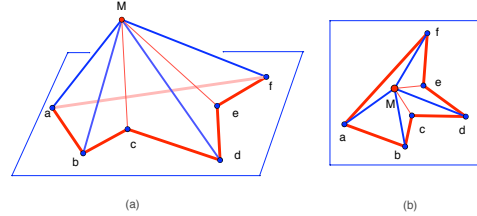


Figure 8: The mountain/valley argument: (a) Cutting just below the vertex of maximum z -coordinate, we get the image in (b).

An alternative proof, not based on Maxwell’s theorem, is deferred to the full paper.

5 The Main Result

Each monotone mechanism induced by a pseudo triangulation can be moved as long as the edge vectors remain acyclic. We complete the proof by showing

how to glue together the trajectories corresponding to motions. A change in acyclicity occurs when two bars become collinear. At that point, we have to recompute a new pseudo triangulation. The following theorem proves that this can be done either with local changes and keeping the same number of vertices of the original polygon, or by decreasing by at least one the number of vertices of the polygon and applying induction.

Theorem 5.1 (Gluing trajectories on the boundary) *On the boundary, one of the following cases can occur:*

1. *Two adjacent edges of the polygon become collinear. In this case, we freeze the joint, eliminating one vertex of the polygon and applying induction to continue.*
2. *Two adjacent added diagonals or one diagonal and an edge of the polygon become collinear. In this case, we perform a flip in the pseudo triangulation, depicted in Fig. 9.*

The above theorem is not stated in full generality, to avoid cluttering the overall picture with details. In particular, several vertices may straighten simultaneously, but the same type of argument would work. The proof and a complete analysis depend on semi-simple frameworks (in particular, polygons), which we omit here for lack of space. Alternatively, we can imagine a *perturbation argument*, used to help with the computation of a new pseudo-triangulation at the boundary case. We must avoid using the same two edges that just became collinear. This can be done by perturbing the joint vertex to a nearby position which would intuitively correspond to what the mechanism would look like “right after” passing through the straightened position. It will no longer be acyclic, and there will be exactly one edge whose removal will make it acyclic again. Then we can use the extension theorem for pseudo triangulations to add a new edge. Perturbing back to the original position of the vertex, the induced mechanism will have two overlapping edges and one of its faces will be a semi-simple pseudo triangle (a particular case of the semi-simple polygons we alluded to in section 2). The correctness of this step depends on the fact that our extended theorem for the existence of infinitesimal motions (section 4) works for semi-simple mechanisms, too.

Theorem 5.2 Termination and number of steps
The convexification of a polygon terminates in at most $\mathcal{O}(n^2)$ steps.

Proof (sketch) At each step, a bar that was dropped from the 1DOF mechanism at the boundary of a motion will never be added back at some later step. Hence at most $\mathcal{O}(n^2)$ steps are needed (as only this many diagonals are available for bracing the polygon).

Theorem 5.3 Main Result: Convexification of Planar Chains with Monotone Motions *Every*

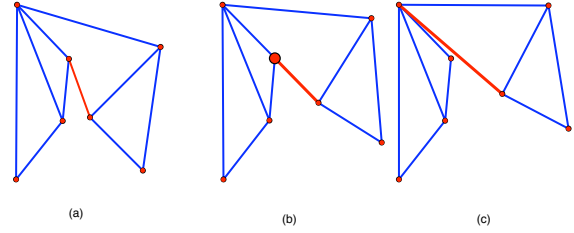


Figure 9: A flip of a diagonal on the boundary.

planar polygon can be convexified with at most $\mathcal{O}(n^2)$ motions. Each motion is induced by a 1DOF monotone mechanism induced by a pseudo triangulation, moved until two of its edges straighten, at which point some diagonals of the mechanism are recomputed. The complete trajectory in configuration space is a sequence of simple curves, each one naturally parametrized by a rotating edge in the work space. A first pseudo triangulation can be computed in (at most) quadratic time, and updated in (at most) linear time per step.

6 Conclusions

We have shown how to compute, algorithmically, a trajectory in the configuration space of a planar linkage, corresponding to a motion that convexifies the polygon without producing self-crossings along the way. The proof is based on a number of novel ideas, most prominently the pseudo triangulations, whose main combinatorial and rigidity theoretic properties have been described.

From the point of view of Rigidity Theory, we have exhibited a class of planar rigid graphs with elegant combinatorial properties. We think that the rigidity theoretic properties of pseudo triangulations will prove to be useful in other applications.

Our work leaves open a number of subproblems for further improvements (such as, the algorithmic problem of finding a pseudo triangulation in subquadratic time, which can most certainly be worked out with some better data structures). But perhaps more important than this particular result is the combinatorial technique based on monotone mechanisms. We conjecture that it can be used to investigate other motion planning problems. In particular, it would be interesting to understand the properties of the non-monotone mechanisms obtained by removing a non-convex hull edge of a pseudo triangulation and their critical configurations.

Acknowledgements

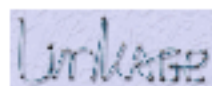
I am grateful to Anna Lubiw and Sue Whitesides, the organizers of the McGill University Workshop on Fold-

ing and Unfolding Problems in Computational Geometry, Barbados 1998, where this work started. Thanks to the Monte Verita 1999 Discrete and Computational Geometry workshop organizers (J.E. Goodman, R. Pollack and E. Welzl) for their invitation. It was there that Günter Rote proposed the use of monotone motions and the linear programming approach to this problem, which provided the inspiration for all my subsequent work. I am grateful to him for his generous sharing of ideas. I acknowledge useful conversations on various aspects of Rigidity Theory with Bob Connelly, Henri Crapo, Ruth Haas, Brigitte and Herman Servatius and Walter Whiteley. Discussions with the other Barbados workshop participants have always been stimulating and I acknowledge them with gratitude. Many thanks to Ciprian Borcea for pointing several connections with the mathematics literature and for stimulating conversations.

References

- [AGP] B. Aronov, J.E. Goodman and R. Pollack. *Convexifying a Planar polygon in R^3* . manuscript, 1999.
- [AB00] P. Agarwal, J. Basch, L. Guibas, J. Hersberger and L. Zhang. *Deformable Free Space Tilings for Kinetic Collision Detection*, Proc. 4th Int. Workshop on Alg. Found. Robotics (WAFR), March 2000, Dartmouth College (to appear).
- [BE99] J. Basch, J. Erickson, L. Guibas, J. Hersberger and L. Zhang. *Kinetic Collision Detection Between Two Simple Polygons*, Proc. 10th ACM-SIAM Symp. Discr. Algorithms (SODA), 1999, pp.102-111.
- [BPRa] S. Basu, R. Pollack and M.-F. Roy. *Computing Roadmaps of Semi-algebraic Sets on a Variety*, JAMS, 13, pp. 55-82, 2000. prelim. version in Proc. 28th ACM Symp. Theory of Computing, pp. 168-173, 1996.
- [BPRb] S. Basu, R. Pollack and M.-F. Roy. *Algorithms in Real Algebraic Geometry*, book, in preparation.
- [BD+98] T. Biedl, E. Demaine, M. Demaine, S. Lazard, A. Lubiw, J. O'Rourke, S. Robbins, I. Streinu, G. Toussaint and S. Whitesides. *On reconfiguring tree linkages: Trees can lock*, Proc. 10th Canad. Conf. Comp. Geom., Montreal, 1998.
- [BD+99] 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*, Proc. 10th ACM-SIAM Symp. Discrete Algorithms (SODA) 1999, pp. 866-867.
- [Bj93] A. Björner, M. Las Vergnas, B. Sturmfels, N. White and G. Ziegler. *Oriented Matroids*, Cambridge University Press, 1993
- [Ca88] J. Canny. *Complexity of Robot Motion Planning*, MIT Press, Cambridge, MA, 1988.
- [COR] R. Cocan and J. O'Rourke. *Polygonal Chains Cannot Lock in 4d*, Proc. 11th Canad.Conf. Comput. Geometry, Vancouver, 1999, pp. 5-8.
- [CDR] R. Connelly, E. Demaine and G. Rote. *Every Polygon Can Be Untangled*, manuscript, 2000
- [CW] H. Crapo and W. Whiteley, *Plane self-stresses and projected polyhedra I: The basic pattern*, Structural Topology 20, 1993, pp. 55-77.
- [Er] P. Erdős. Problem number 3763. Am. Math. Monthly, 46:627, 1935.
- [GP] J.E. Goodman and R. Pollack. *Allowable sequences and Order Types in Discrete and Computational Geometry*, in *New Trends in Discrete and Computational Geometry*, J. Pach (ed.), Springer Verlag, 1993
- [GSS] J. Graver, B. Servatius and H. Servatius. *Combinatorial Rigidity*, Amer. Math. Soc., Graduate Studies in Mathematics vol.2, 1993.
- [HA] L. Han and N. Amato. *A Kinematics-Based Probabilistic Roadmap Method for Closed Chain Systems*. Proc. 4th Int. Workshop on Alg. Found. Robotics (WAFR), March 2000, Dartmouth College (to appear).
- [HJW] J. Hopcroft, D. Joseph and S.H. Whitesides. *On the movement of robot arms in 2-dimensional bounded regions*, SIAM J. Comput., 14:315-333, 1985.
- [JP] D. Joseph and W.H. Platinga. *On the complexity of reachability and motion planning questions*, Proc. Symp. Comp. Geom., pp. 62-66, 1985.
- [KM95] M. Kapovich and J. Millson. *On the moduli spaces of polygons in the Euclidian plane*, Journ. of Diff. Geometry, vol. 42, N 1, 1995, pp. 133-164.
- [KM98] M. Kapovich and J. Millson. *Universality Theorems for configurations spaces of planar linkages*, preprint, 1998.
- [Ka] L. Kavraki, P. Svestka, J.C. Latombe and M. Overmars. *Probabilistic roadmaps for path planning in high-dimensional configuration spaces*. IEEE Trans.Robot. Automat., 12(4): 566-580, Aug. 1996.
- [LW] W.J. Lenhart and Sue H. Whitesides. *Reconfiguring Closed Polygonal Chains in Euclidian d-space*. Discr. Comp. geometry 13:123-140, 1995.
- [LY] L. Lovasz and Y. Yemini. *On generic rigidity in the plane*, SIAM J. Algebraic and Discrete Methods, 3, no. 1, 91-98.
- [PV] M. Pocchiola and G. Vegter. *Computing the Visibility Graph via Pseudo-Triangulations*, Proc. 11th ACM Symp. Comput. Geometry, 1995, pp. 248-257.

- [Re] A. Recski. *Matroid Theory and Its Applications*, Springer Verlag, 1989.
- [SS] J. Schwartz and M. Sharir. *On the piano mover's problem, II. General techniques for computing topological properties of real algebraic manifolds*. Adv. in Appl. Math., 4:298-351, 1983.
- [St] I. Streinu. *Clusters of Stars*, Proc. ACM Symp. Comp. Geometry, Nice, 1997, pp. 439-441.
- [To99] G. Toussaint. *The Erdős-Nagy Theorem and its Ramifications*, Proc. Canad. Conf. Comp. Geom., 1999.
- [W96] W. Whiteley. *Some Matroids from Discrete Applied Geometry*, Contemporary Mathematics, vol. 197, pp. 171-311, Amer. Math. Soc. 1996.
- [W97] W. Whiteley. *Rigidity and Scene Analysis*, in *Handbook of Discrete and Computational Geometry*, (J.E. Goodman, J. O'Rourke, eds.), 1997, pp. 893-916.
- [W92] S.H. Whitesides. *Algorithmic issues in the geometry of planar linkage movement*, Austral. Comp. J., Spec. Issue on Algorithms, 24(2):42-50, 1992.



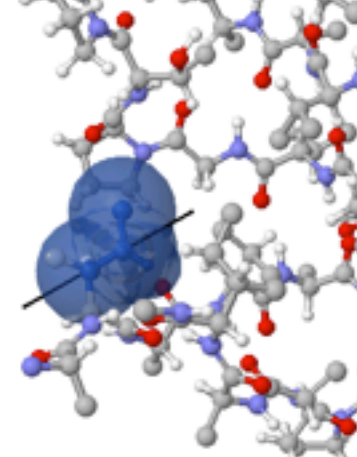
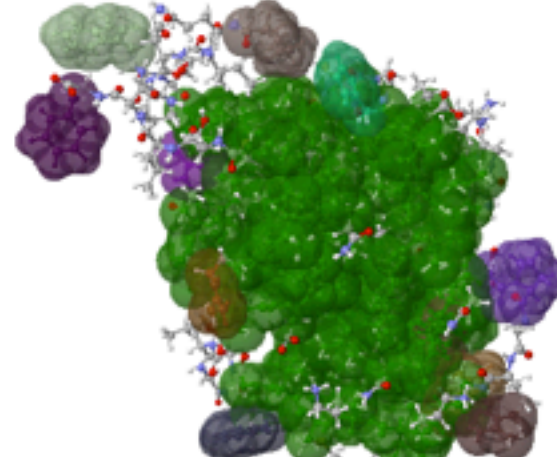
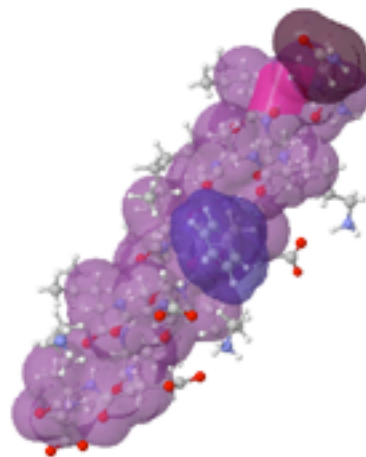
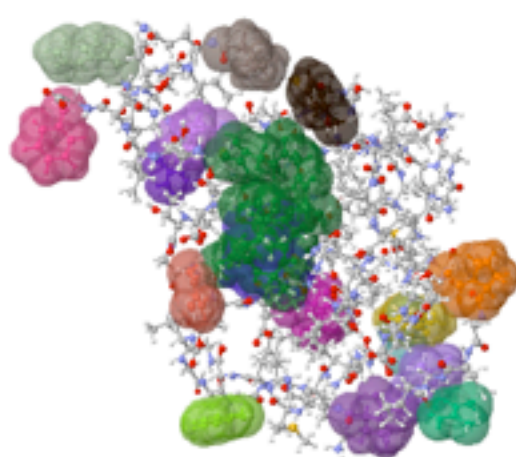
UMassAmherst



SMITH COLLEGE

KINARI Software

for KINematics And Rlgidity analysis of proteins

[Linkage Lab](#)[KINARI Home](#)[KINARIWeb](#)[Educational](#)[Downloads](#)[About](#)

Try KINARIWeb

Enter 4-digit PDB code: (e.g. 1PEF)

[Reset](#)[Run](#)

Request access

Funding provided by
NSF CCF-0728783
NSF DMS-0714934
DARPA 23 Mathematical Challenges

KINARIWeb

for KINematics And RIgidity analysis of proteins | [version info](#)

[KINARIWeb](#)[KINARI Home](#)[Documentation](#)[Report Bug](#)

↑ Input

Input a PDB in one of the following ways

- Enter 4-digit PDB code: (e.g. 1PEF) - OR -
- Upload a PDB file: no file selected - OR -
- [Create a new pdb file](#)



↓ Options

KINARIWeb

for KINematics And Rlgidity analysis of proteins | [version info](#)

[KINARIWeb](#)[KINARI Home](#)[Documentation](#)[Report Bug](#)

↑ Input

Your input: 1hhp.pdb

[Run](#)

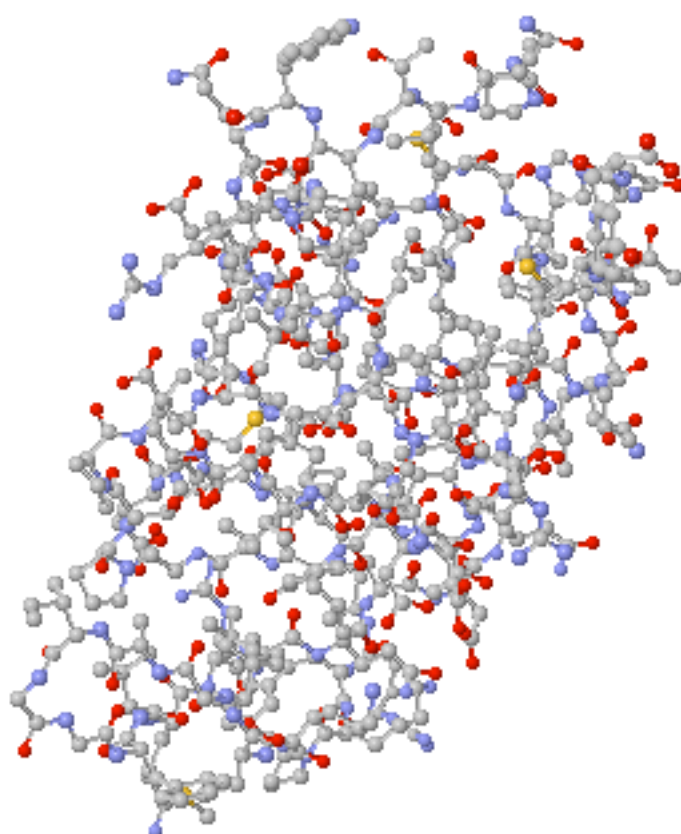
Input a PDB in one of the following ways

- Enter 4-digit PDB code: (e.g. 1PEF) [Retrieve](#) - OR -
- Upload a PDB file: [Choose File](#) no file selected - OR -
- [Create a new pdb file](#)

↓ Options

Visualization [\[?\]](#)

1hhp.pdb



↑ Input

Your input: ☒ 1hhp.pdb



Run

Input a PDB in one of the following ways

- Enter 4-digit PDB code: (e.g. 1PEF) - OR -
- Upload a PDB file: no file selected - OR -
- [Create a new pdb file](#)

↑ Options

:: PDB Parameters ::

Chain ID: Model NUM: [\[?\]](#)

:: Hydrogen Bonds ::

- ☒ add hydrogens [\[?\]](#)
- ☒ add hydrogen bonds [\[?\]](#)

energy cutoff [\[?\]](#)

Hydrogen Bond Strength: [\[?\]](#)

:: Hydrophobics ::

- ☒ add hydrophobics

distance cutoff [\[?\]](#)

Hydrophobics Strength: [\[?\]](#)

:: Disulfide Bonds ::

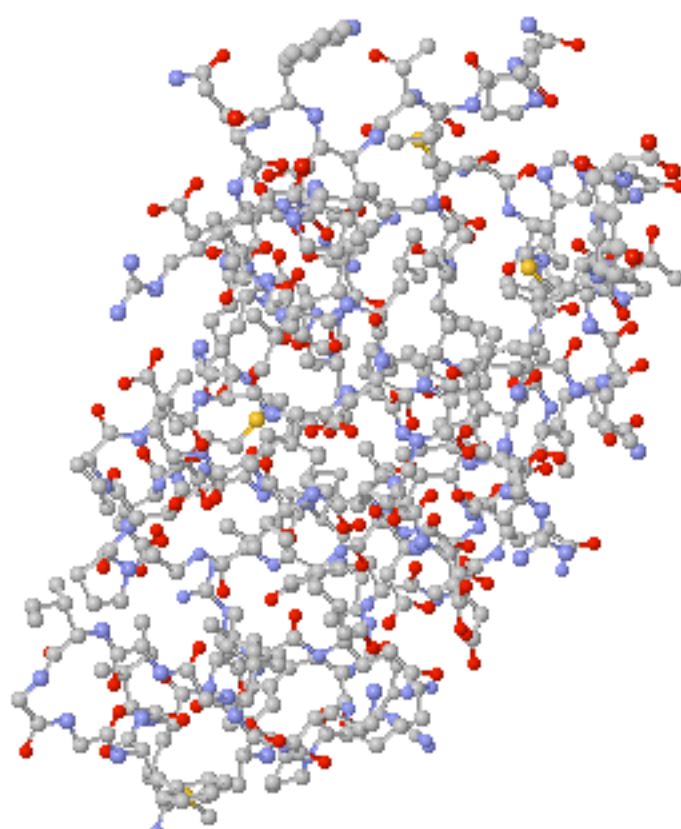
Disulfide Strength: [\[?\]](#)

:: Modeling Type ::

Cluster representation [\[?\]](#) ☒ KINARI ☐ ASU FIRST

Visualization [\[?\]](#)

1hhp.pdb



KINARIWeb

for KINematics And Rlgidity analysis of proteins | [version info](#)

[KINARIWeb](#)[KINARI Home](#)[Documentation](#)[Report Bug](#)

↓ Input

Your input: 1hhp.pdb

[Run](#)

↓ Options

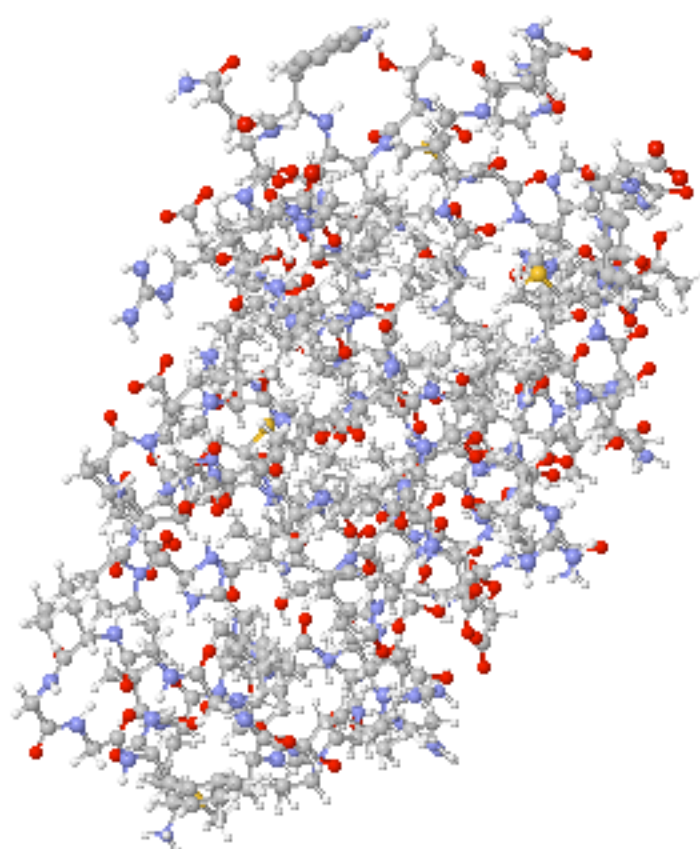
Visualization [\[?\]](#)

1hhp.pdb

Result Summary [\[?\]](#)

Num of clusters	245	ID of largest cluster	33	Size of largest cluster	717
Num of hinges	247	Num of bars	35	Num of user-defined constraints	0

Click "load clusters" button to view rigid clusters of the protein.



Jmol

[load clusters](#)

Orientation: [Save](#) [Restore](#) Cartoon: ☐ on/off

For more viewing options, right click on the above window. (See [Jmol Documentation](#))

↓ Output [\[?\]](#)

KINARIWeb

for KINematics And RIgidity analysis of proteins | [version info](#)

[KINARIWeb](#)[KINARI Home](#)[Documentation](#)[Report Bug](#)

↓ Input

Your input: 1hbp.pdb

[Run](#)

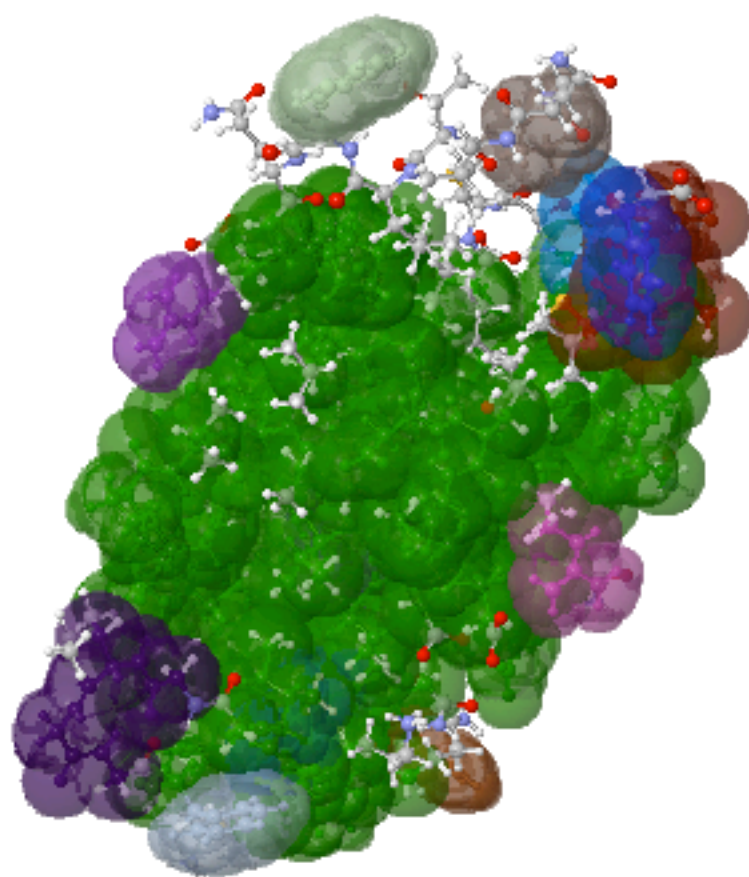
↓ Options

Visualization [\[?\]](#)

1hbp.pdb

Result Summary [\[?\]](#)

Num of clusters	245	ID of largest cluster	33	Size of largest cluster	717
Num of hinges	247	Num of bars	35	Num of user-defined constraints	0



↑ Clusters [\[?\]](#)

Click on atoms to find out their associate clusters ID

Cluster Options

[display all clusters](#)

Display only clusters larger than atoms.

Display in isolation cluster ID

Surface Options

[display all surfaces](#)[hide all surfaces](#)

Show surfaces for clusters larger than atoms.

Turn on the surface of cluster ID

Turn off the surface of cluster ID

↓ Hinges [\[?\]](#)

↓ Bars [\[?\]](#)

Jmol

[default view](#)

Orientation: [Save](#) [Restore](#) Cartoon: ☐ on/off

For more viewing options, right click on the above window. (See [Jmol Documentation](#))

↓ Output [\[?\]](#)

↓ Input

Your input: 

Run

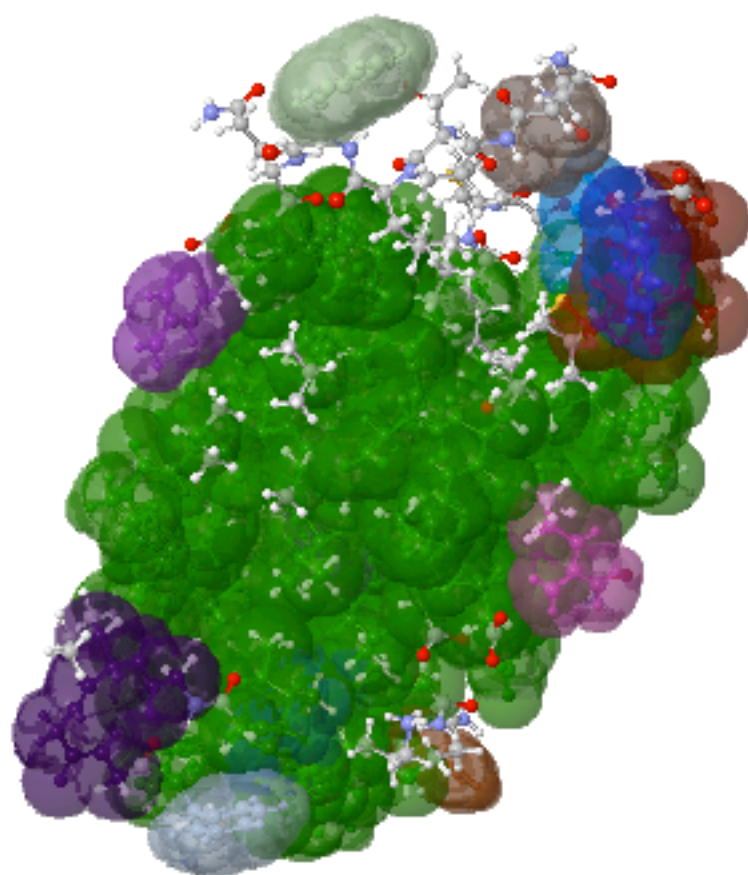
↓ Options

Visualization [\[?\]](#)

1hhp.pdb

Result Summary [\[?\]](#)

Num of clusters	245	ID of largest cluster	33	Size of largest cluster	717
Num of hinges	247	Num of bars	35	Num of user-defined constraints	0



↑ Clusters [\[?\]](#)

Click on atoms to find out their associate clusters ID

Cluster Options

Display only clusters larger than atoms.Display in isolation cluster ID

Surface Options

Show surfaces for clusters larger than atoms.Turn on the surface of cluster ID Turn off the surface of cluster ID

↓ Hinges [\[?\]](#)

↓ Bars [\[?\]](#)

Jmol

Orientation: Cartoon: ☐ on/offFor more viewing options, right click on the above window. (See [Jmol Documentation](#))

↑ Output [\[?\]](#)

:: 1hhp ::

Job summary	summary.txt
Configuration file	config.xml
List of covalent bonds	1hhp_cov.out
List of double covalent bonds	1hhp_doublecov.out
List of hydrogen bonds	1hhp_hbonds.out
List of hydrophobes	1hhp_hphobes.out
Processed PDB	1hhp_processed.pdb
Rigid clusters, hinges and bars	1hhp_postPG_BBH.xml

↓ Input

Your input: 1hhp.pdb 

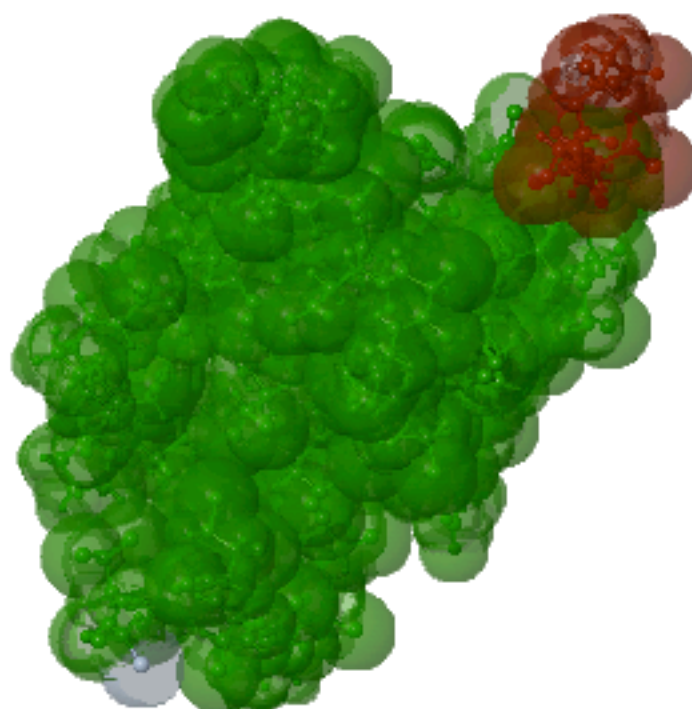
↓ Options

Visualization [\[?\]](#)

1hhp.pdb

Result Summary [\[?\]](#)

Num of clusters	245	ID of largest cluster	33	Size of largest cluster	717
Num of hinges	247	Num of bars	35	Num of user-defined constraints	0



↑ Clusters [\[?\]](#)

Click on atoms to find out their associate clusters ID

Cluster Options

Display only clusters larger than atoms.

Display in isolation cluster ID

Surface Options

Show surfaces for clusters larger than atoms.

Turn on the surface of cluster ID

Turn off the surface of cluster ID

↓ Hinges [\[?\]](#)

↓ Bars [\[?\]](#)

Jmol



Orientation: Cartoon: ☐ on/off

For more viewing options, right click on the above window. (See [Jmol Documentation](#))

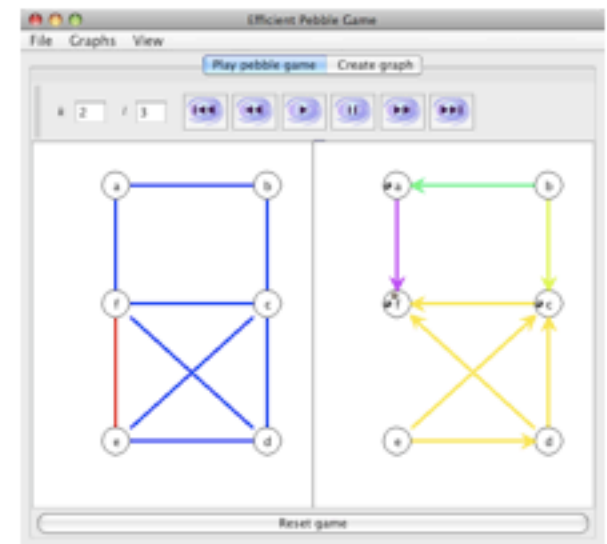
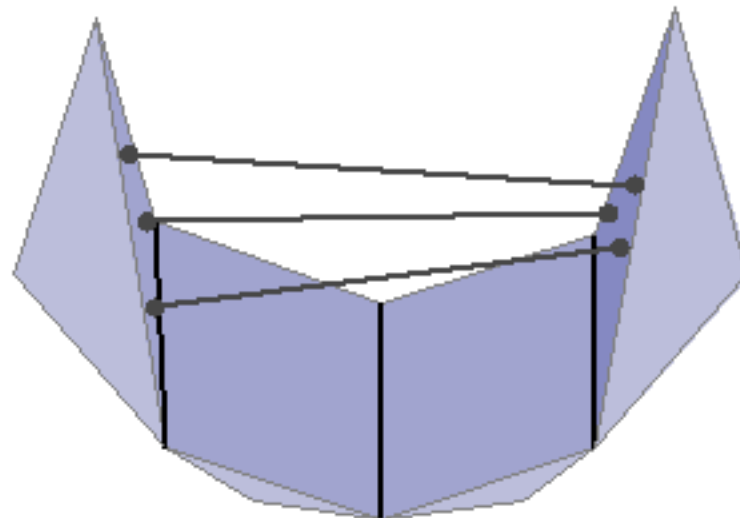
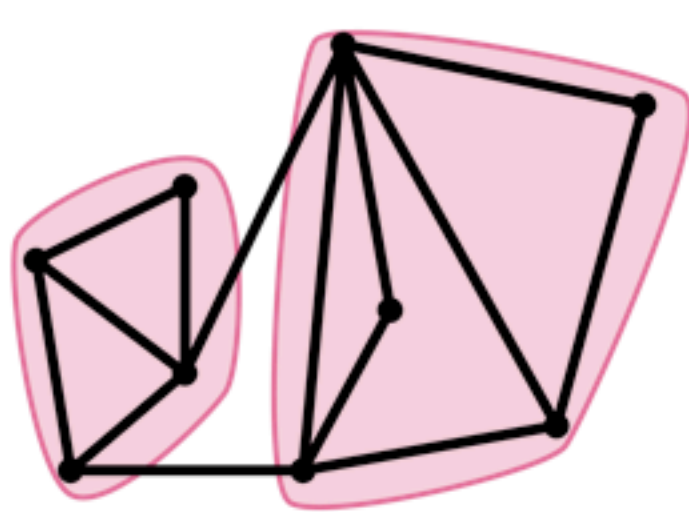
↑ Output [\[?\]](#)

:: 1hhp ::

Job summary	summary.txt
Configuration file	config.xml
List of covalent bonds	1hhp_cov.out
List of double covalent bonds	1hhp_doublecov.out
List of hydrogen bonds	1hhp_hbonds.out
List of hydrophobes	1hhp_hphobes.out
Processed PDB	1hhp_processed.pdb
Rigid clusters, hinges and bars	1hhp_postPG_BBH.xml

Rigidity and Pebble games

Audrey Lee, Ileana Streinu, Louis Theran



[Home](#)
[Rigidity](#)
[Pebble Games](#)
[Sparsity](#)
[Video](#)
[Bibliography](#)

Overview

How many pair-wise distances must be prescribed between an unknown set of points, and how should they be distributed, to determine only a discrete set of possible solutions? These questions, and related generalizations, are central in a variety of applications ranging from engineering to protein structure determination to origami folding. **Combinatorial rigidity** answers questions on generic structures via graph properties.

This web site is devoted to demonstrations of our interactive software for **Pebble Game Algorithms** for analyzing the rigidity and flexibility of a variety of structures.

Contents

- ↓ [Rigidity Models and Questions.](#)
- ↓ [Pebble Games for Rigidity.](#)
- ↓ [Pebble Games for Sparse Graphs.](#)
- ↓ [Bibliography.](#)

Research funded by the NSF under grants NSF CCF-0430990 and NSF-DARPA CARGO CCR-0310661.

Rigidity Models and Questions

Audrey Lee, Ileana Streinu, Louis Theran

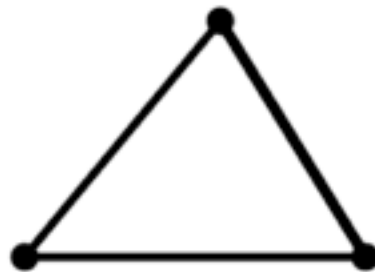
Home	Rigidity	Pebble Games	Sparsity	Video	Bibliography	
------	----------	--------------	----------	-------	--------------	--

↑ Bar-and-joint rigidity in the plane

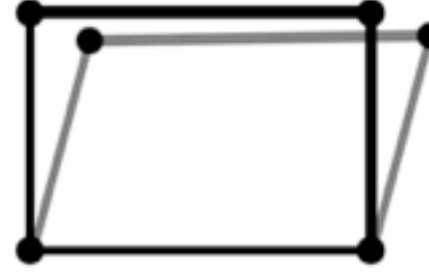
Top
2D bar-joint
Rigidity questions
3D Body-bar
2D bar-slider

Planar bar-and-joint frameworks: A planar bar-and-joint framework is a structure made of **fixed-length bars** connected by **universal joints**, with full rotational freedom. The allowed motions are those that preserve the lengths and connectivity of the bars. (The bars are allowed to "pass through" each other.) When the only allowed motions are those that preserve the distance between *every* pair of joints, the framework is said to be **rigid**. Otherwise it is **flexible**.

A framework with three bars making a triangle must be rigid, since all the possible distances are fixed by bars. A rigid framework in the plane has three trivial degrees of freedom: one rotational and two translational.



Rigid



Flexible

On the other hand, a four-bar mechanism has a non-trivial motion, which implies that it is flexible. Since this motion is allowed even when an edge of the framework is pinned down to factor out trivial degrees of freedom, this motion is called an **internal motion**.

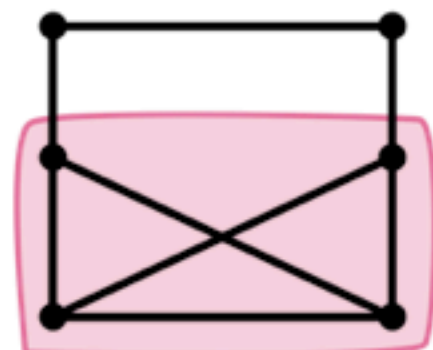
Laman's theorem and combinatorial rigidity: The most fundamental question we can ask about a framework is whether it is rigid or flexible. For almost all planar bar-and-joint frameworks, Laman's foundational theorem gives a *combinatorial* answer. We associate each framework with a graph that has one vertex for each joint and one edge for each bar. (Notice that many different frameworks can have the same graph.)

Theorem (Laman): Let G be a graph with n vertices and m edges. G is the graph of a generic minimally rigid framework if and only if: any subset of n' vertices in G spans at most $2n'-3$ edge; and $m=2n-3$.

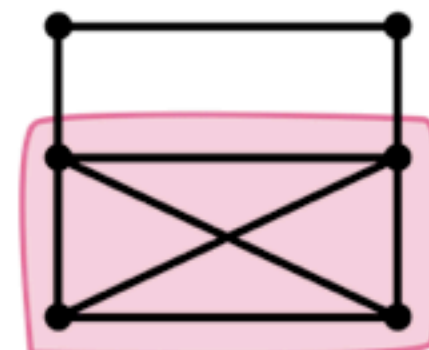
The genericity condition in Laman's theorem holds for almost all choices for the lengths of the bars for a given graph G . Genericity is a somewhat technical concept, but it can be intuitively thought of as playing the role of "general position" assumptions that are a paradigmatic part of computational geometry.

Rigid components and redundant edges: Laman's theorem gives a combinatorial method for analyzing the rigidity of frameworks by counting the number of bars. For example, we can see that the triangle is minimally rigid, since $3=2 \times 3 - 3$ and all the subgraphs satisfy the Laman count. The four-bar mechanism must be flexible, since $4 < 2 \times 4 - 3 = 5$.

Flexible frameworks still may contain substructures that move rigidly. We call a set of vertices a **rigid component** if it is inclusionwise maximal with the property all of the vertices in the set move rigidly with respect to each other. Since the endpoints of every bar move rigidly with respect to each other, we see that every joint must be in some rigid component. In the example of the four-bar mechanism, each bar forms one rigid component.



Rigid components



Rigid components and redundant edges

Rigid components of flexible frameworks need not be (and in general, will not be) single edges. In the examples above, both frameworks have one internal motion, similar to that of the four-bar mechanism. However, both have a larger rigid component, highlighted in red. The example on the left, like the four-bar mechanism, does not have enough edges to be rigid.

On the right is a framework that does have enough bars to be rigid, but it violates the Laman counts on the set of vertices shown in red. This means that there is some edge that can be removed from the framework without changing its rigidity properties. We call such an edge **redundant**, and such a set of vertices **over constrained**. Observe that there is not a single edge that is redundant in this case: removing any of the edges spanned by the highlighted set of vertices results in a framework with the same rigid components as the original. We also see here that rigid components need not be minimally rigid substructures of a framework.

Rigidity Models and Questions

Audrey Lee, Ileana Streinu, Louis Theran

Home	Rigidity	Pebble Games	Sparsity	Video	Bibliography	
------	----------	--------------	----------	-------	--------------	--

↓ Bar-and-joint rigidity in the plane

↓ Rigidity analysis questions

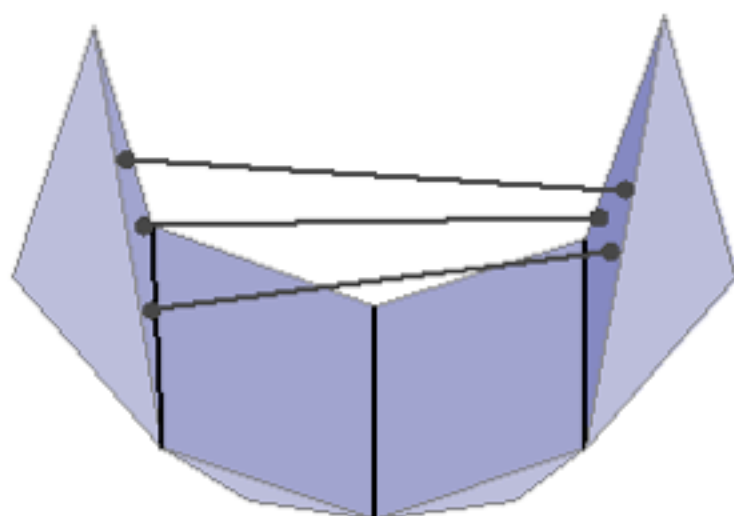
↑ 3D Body-bar rigidity

Top
2D bar-joint
Rigidity questions
3D Body-bar
2D bar-slider

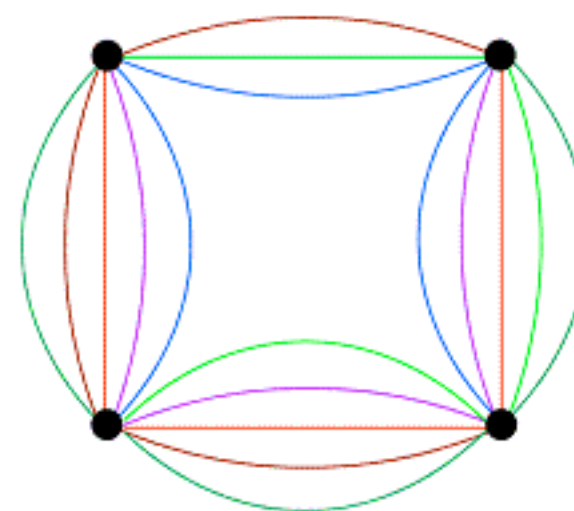
Motivation: Bar-and-joint rigidity is elegant and can be used to model many structures. However, a notable drawback (and major open problem!) is the lack of an extension of Laman's theorem to dimension 3. We now introduce another commonly used rigidity model for which there is such a theorem.

Body-bar-hinge frameworks: A **body-bar framework** is a 3D structure made of **rigid bodies** connected by **fixed-length bars** attached to **universal joints** with full rotational freedom. (As in the case of planar bar-joint frameworks, the bodies and bars may "pass through" each other for the purpose of rigidity analysis.) A body-bar framework is **rigid** if the only motions which maintain the lengths of all of the bars are trivial rigid motions and **flexible** otherwise.

A **body-bar-hinge framework** is a body-bar framework that may also have **hinges** between two of the bodies.



A rigid body-bar-hinge framework



The associated multigraph

The combinatorial model of a body-bar-hinge framework is a multigraph with one vertex for each body, one edge for each bar between two bodies, and five edges for each hinge between two bodies. Tay proved the following Laman like result for body-bar-hinge frameworks.

Theorem (Tay): Let G be a graph with n vertices and m edges. G is the graph of a generic minimally rigid body-bar-hinge framework if and only if: any subset of n' vertices in G spans at most $6n'-6$ edges; and $m=6n-6$.

We point out the similarities between the counting conditions from Tay's theorem and Laman's theorem. Intuitively, we have $6n-6$ instead of $2n-3$ because each rigid body has 6 trivial degrees of freedom to start with, and each bar removes at most one degree of freedom.

Pebble games for rigidity

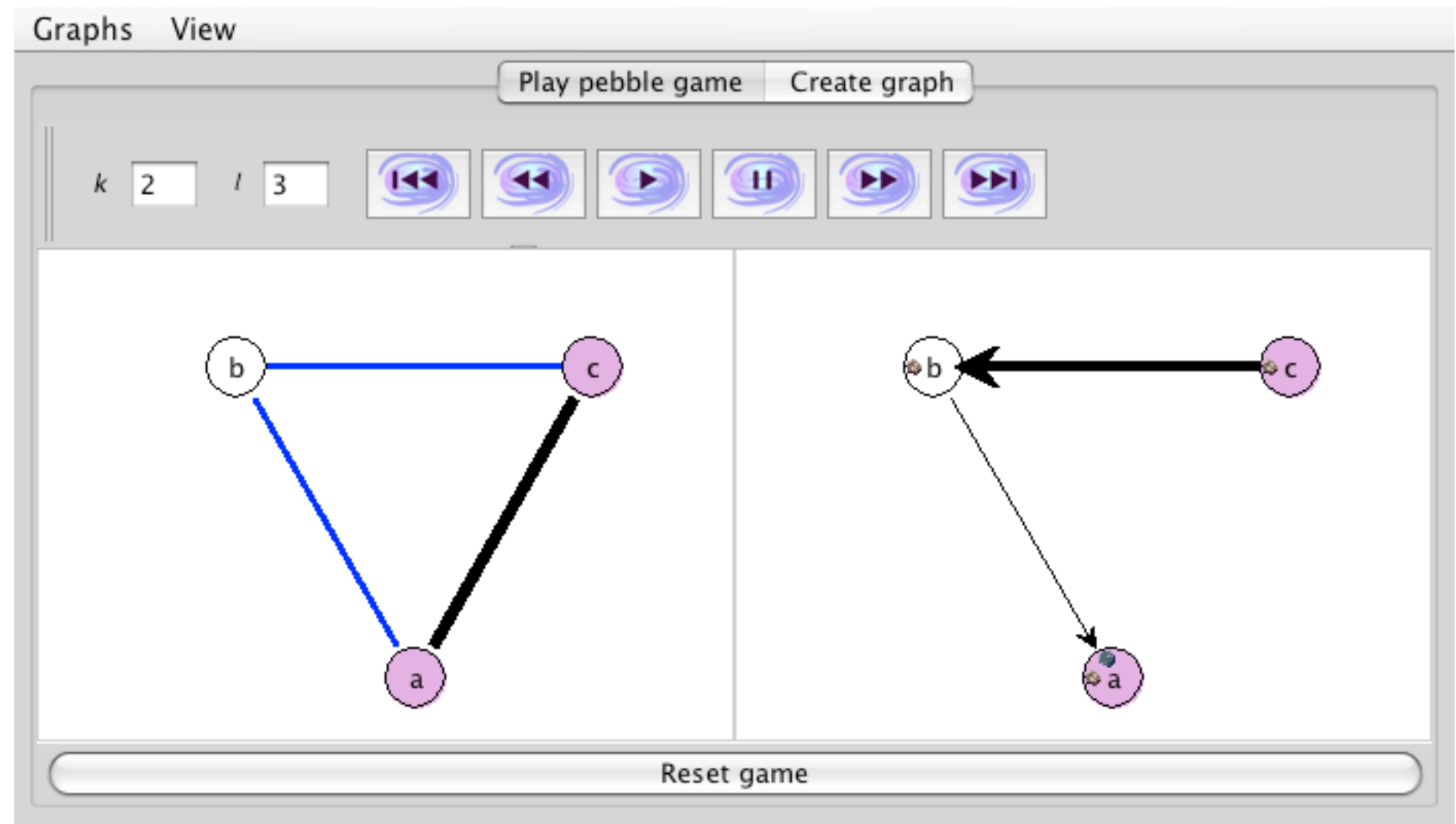
Audrey Lee, Ileana Streinu, Louis Theran

Home	Rigidity	Pebble Games	Sparsity	Video	Bibliography	
------	----------	--------------	----------	-------	--------------	--

↑ The basic pebble game

Top
Basic pebble game
Rigid components
3D Body-bar
2D bar-slider

Checking the Laman counts: The **basic pebble game**, first introduced by Jacobs and Hendrickson, is an elegant algorithm for the **Decision** rigidity analysis question. Press the "play" in the applet below to see it run on the triangle, which we know to be minimally rigid.



Instructions:

- To select a built in graph, go to the Graphs menu in the applet.
- To make your own graph, click on the create graph tab.
- Use the controls to animate, pause, and step the game.

Playing the game: The pebble game can be described in terms of its initial configuration and the allowed moves. The game is played on a directed graph (the right-hand panel in the applet), which serves as the "board".

Initial configuration: Each vertex starts with two pebbles on it, and there are no edges.

Add edge move: If i and j are vertices with two pebbles on each, add the edge ij (oriented toward j) and pick up one of the pebbles from i .

Pebble slide move: If ij is an edge in the pebble game's graph and there is a pebble on j , reverse ij and move the pebble from j to i .

Edges may be added exactly when the extended graph will satisfy the Laman counts.

Theorem: The graphs that can be constructed using the basic pebble game are exactly those that satisfy the Laman counts. The order in which the edges are added does not matter.

To use the pebble game for the **Decision** problem the strategy is then to try and collect four pebbles on the ends of each of the inputs edges in an arbitrary order. Since the pebbles can be collected with DFS, this algorithm is polynomial time.

Rejecting redundant edges: When there are redundant edges in the input, it is impossible to collect the four pebbles required to add the edge. Observe in the example below that once there are 5 edges spanned by the bottom vertices, it is impossible to collect four pebbles on the edges of the sixth edge in this subgraph, so the edge is rejected.

Pebble games for rigidity

Audrey Lee, Ileana Streinu, Louis Theran

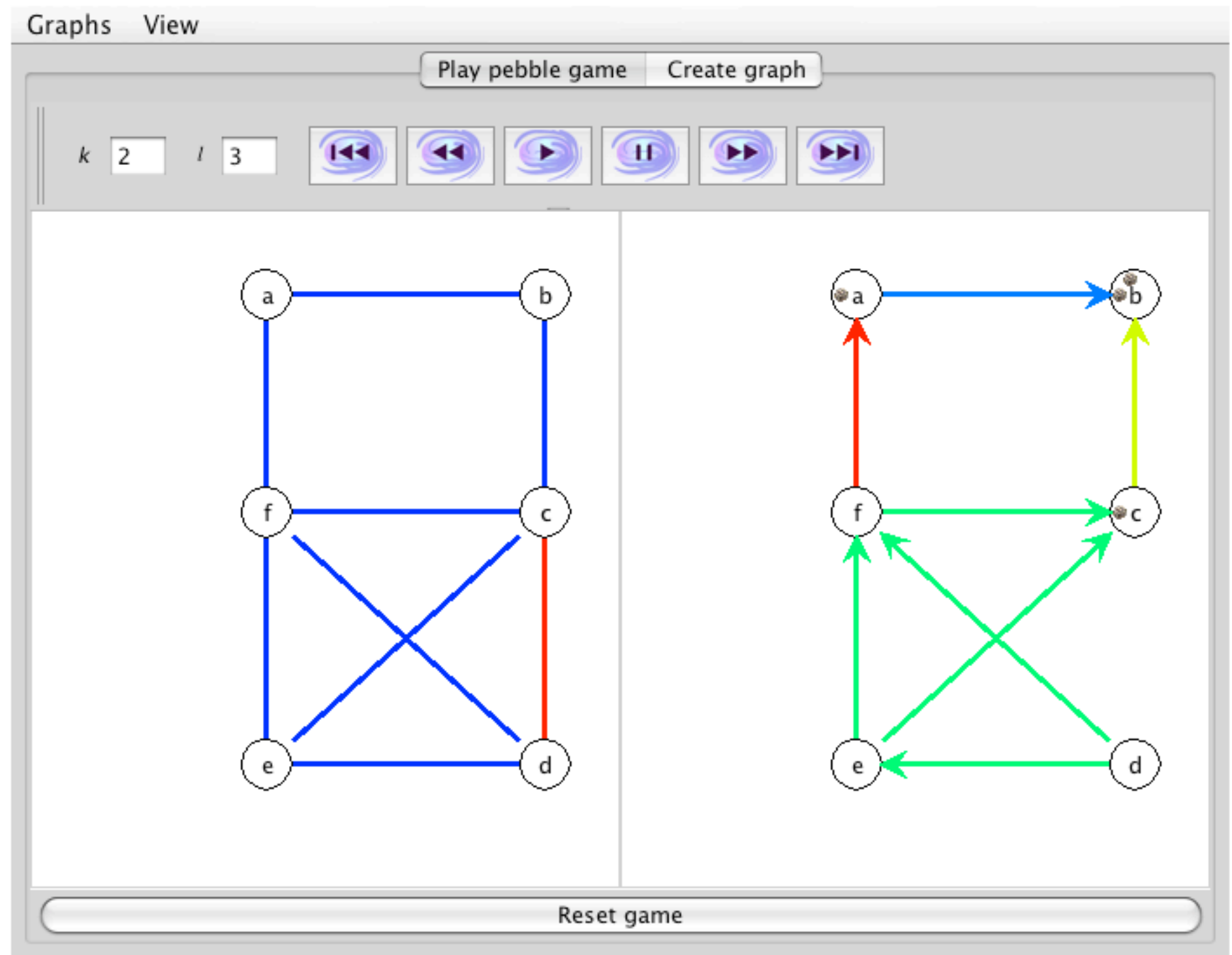
[Home](#)[Rigidity](#)[Pebble Games](#)[Sparsity](#)[Video](#)[Bibliography](#)

↓ The basic pebble game

↑ The rigid components pebble game

[Top](#)
[Basic pebble game](#)
[Rigid components](#)
[3D Body-bar](#)
[2D bar-slider](#)

Finding and maintaining rigid components: We extended the pebble game to find and update the rigid components of the pebble game graph each time an edge is added. In the example below, each rigid component is shown in a different color on the right.



The components pebble game solves the **Components** rigidity question. Maintaining components, with an appropriate data structure we call **union/pair-find** also allows redundant edges to be rejected in constant time. This improves the $O(n^3)$ running time of the basic pebble game on dense inputs to $O(n^2)$.

Pebble games for rigidity

Audrey Lee, Ileana Streinu, Louis Theran

Home	Rigidity	Pebble Games	Sparsity	Video	Bibliography	
------	----------	--------------	----------	-------	--------------	--

↓ The basic pebble game

↓ The rigid components pebble game

↑ The pebble game for body-bar rigidity

Top
Basic pebble game
Rigid components
3D Body-bar
2D bar-slider

The pebble game for Tay's counts: Our pebble game for body-bar rigidity has a similar structure to the pebble game for 2D baj-and-joint rigidity. We describe it in terms of the initial configuration and the allowed moves, showing the differences in bold.

- Initial configuration:** Each vertex starts with **six** pebbles on it, and there are no edges.
- Add edge move:** If i and j are vertices with at **least seven pebbles between them**, add the edge ij (oriented toward j) and pick up one of the pebbles from i .
- Pebble slide move:** If ij is an edge in the pebble game's graph and there is a pebble on j , reverse ij and move the pebble from j to i .

We see that the rules for moving pebbles are the same, and that the different counts are captured by the number of pebbles that start on each vertex and the number of pebbles required to add an edge.

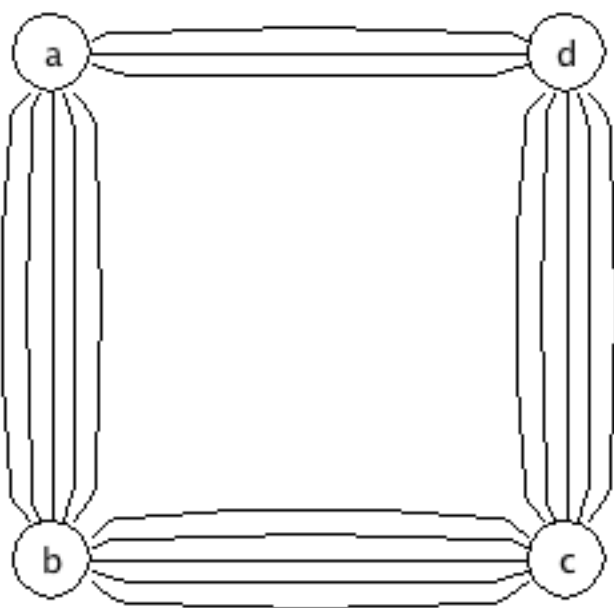
All the variants of the pebble game work with the $6n-6$ counts. Here we show the components pebble game for body-bar.

Graphs View

Play pebble game Create graph

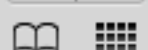
k 6 l 6





Reset game

Note that in body-bar rigidity, not every vertex and edge needs to be in a rigid component. This example is minimally rigid, but has no rigid components if any of the edges are removed. Observe that as the game is being played, none of the edges on the right are colored (to indicate that they are in a component) until the last step.



Bibliography

Audrey Lee, Ileana Streinu, Louis Theran

Home	Rigidity	Pebble Games	Sparsity	Video	Bibliography
------	----------	--------------	----------	-------	--------------

arxiv
doi
.bib
abs

Pebble game algorithms and sparse graphs
Audrey Lee and Ileana Streinu
Discrete Mathematics, 308(8):1425-1437, 2008.

arxiv
.bib
abs

Combinatorial Genericity and Minimal Rigidity
Ileana Streinu and Louis Theran
To appear in Proceedings of the 24th Symposium on Computational Geometry. 2008.

arxiv
.bib
abs

Graded Sparse Graphs and Matroids
Audrey Lee, Ileana Streinu, and Louis Theran
Journal of Universal Computer Science, 13(10), 2007.

.pdf
.bib
abs

The Slider-pinning Problem
Audrey Lee, Ileana Streinu, and Louis Theran
Proc. 19th Canadian Conference on Computational Geometry
Carleton University, Ottawa, Ontario, Canada, Aug. 20-22, 2007.

arxiv
.bib
abs

Sparsity-certifying Graph Decompositions
Ileana Streinu and Louis Theran
Manuscript, 2007.

arxiv
.bib
abs

Sparse Hypergraphs and Pebble Game Algorithms
Ileana Streinu and Louis Theran
European Journal of Combinatorics, special issue for the Proceedings of Oriented Matroids (OM'05), 2007 (to appear).

.pdf
.bib
abs

Finding and Maintaining Rigid Components
Audrey Lee, Ileana Streinu, and Louis Theran
Proc. 17th Canadian Conference on Computational Geometry
Univ. of Windsor, Ontario, Canada, Aug. 10-12, 2005.