

12-1-2000

Combinatorial Approach to Planar Non-Colliding Robot Arm Motion Planning

Ileana Streinu
Smith College, istreinu@smith.edu

Follow this and additional works at: https://scholarworks.smith.edu/csc_facpubs



Part of the [Computer Sciences Commons](#)

Recommended Citation

Streinu, Ileana, "Combinatorial Approach to Planar Non-Colliding Robot Arm Motion Planning" (2000).
Computer Science: Faculty Publications, Smith College, Northampton, MA.
https://scholarworks.smith.edu/csc_facpubs/287

This Conference Proceeding has been accepted for inclusion in Computer Science: Faculty Publications by an authorized administrator of Smith ScholarWorks. For more information, please contact scholarworks@smith.edu

A Combinatorial Approach to Planar Non-colliding Robot Arm Motion Planning

Ileana Streinu *

Department of Computer Science

Smith College

Northampton, MA 01063, USA

streinu@cs.smith.edu

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

Abstract

We propose a combinatorial approach to plan non-colliding motions for a polygonal bar-and-joint framework. Our approach yields very efficient deterministic algorithms for a category of robot arm motion planning problems with many degrees of freedom, where the known general roadmap techniques would give exponential complexity. It is based on a novel class of one-degree-of-freedom mechanisms induced by pseudo triangulations of planar point sets, for which we provide several equivalent characterizations and exhibit rich combinatorial and 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 in the convexification motion consists in moving a pseudo-triangulation-based mechanism along its unique trajectory in configuration space until two adjacent edges align. At that point, a local alteration restores the pseudo triangulation. The motion continues for $\mathcal{O}(n^2)$ steps until all the points are in convex position.

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 or flexible polygonal chains). All of our constructions are elementary and are based on a novel class of planar embedded graphs called *pseudo-triangulations* which possess rich combinatorial properties. We use techniques from Rigid-

ity 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 continuously reconfiguring a simple planar polygon to any other planar configuration with the same edge-lengths, 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.

The paper is organized as follows. In the remaining of this introduction, we give an informal high-level view of the result and its connection with previous work. Preliminary concepts are defined in section 2. Section 3 contains several combinatorial characterizations of pseudo triangulations, as well as a sketch of algorithms for computing them. Section 4 contains the rigidity theoretic results on pseudo-triangulations. Section 5 contains the description of the global convexification motion and complexity analysis. We conclude with some suggestions for further research.

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 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 *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-self-intersecting path with n vertices, and a *closed chain* is a simple 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 con-

*Supported by NSF grant CCR-9731804.

vex polygon. Other types of frameworks of interest in this paper include semi-simple polygons and pseudo-triangulations, defined below.

The Carpenter’s Rule Problem: *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 [12] 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 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 [32], [10], [4] and [5]. In practical applications, probabilistic roadmap heuristics were used instead, see [24], [18]. Under various conditions, problems about reconfiguration of linkages range in complexity from polynomial ([26]) to NP- and even PSPACE-hard, see [19], [38], [21].

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 [14]. See Toussaint [34] 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 [26] 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 [22] aim at understanding the topology of the configuration space of closed chains, but they allow crossings. The universality result for mechanical linkages ([20], [23]) holds for the general self-intersecting case.

Studying reconfigurations of linkages with non crossing motions has received a recent impetus in [28], and results on planar linkages using spatial motions ([7], [1]), trees, 3 and higher-dimensional linkages ([6], [11]) have followed. The Carpenter’s Rule question, raised in the 1970’s in the Topology community by G. Bergman, U. Grenander, S. Schanuel (cf. [25]) and independently in the early 1990’s in the Computer Science community by W. Lenhart, S. Whitesides, J. Mitchell, seems to have first appeared in print in [26] and [25]. It was recently settled by Connelly, Demaine and Rote [12]: all chains can be convexified, all linkages can be straightened. Their approach is to first prove (using linear programming duality and Maxwell’s theorem, using a technique originating in Crapo and Whiteley [13] and Whiteley [35]) that there always exists a local, infinitesimal motion which never decreases any interdistances. 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.

Our Results. We strengthen and provide an algorithmic extension of the above mentioned result. While they have proven that the configuration space of planar non-crossing chains with the same orientation is connected, we show how to algorithmically compute a path, consisting of a finite number of finitely-described arcs, between any two such configurations. 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 [29] in their study of the visibility complex and recently applied to kinetic geometric algorithms ([2], [3]).

Novelty. Our approach is based on the idea of abstracting oriented-matroidal properties that hold throughout a portion of a continuous motion. 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 [12] by showing the non-existence of self-stress in bar-and-strut frameworks of a more complex structure than those arising from 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 factored out), is a finite sequence of curve segments (arcs) connecting continuously at their endpoints.

Each arc corresponds to the unique free motion of 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. It 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 two adjacent edges align, at which moment it ceases to be a pseudo-triangulation. We either freeze a joint (if the aligned edges belong to the polygon) and locally patch a pseudo-triangulation for a polygon with one less vertex, or otherwise perform a local flip of the added diagonals.

There are many ways to construct the initial pseudo-triangulation (e.g. using an adaptation of the greedy flip algorithm of Pocchiola and Vegter [29]) or to patch it at the alignment points. For the sake of the analysis, we use a canonical way which helps us to keep track of a global integer valued weight function $f(n)$. Each

alignment step decreases $f(n)$ by at least one unit. Hence the algorithm stops in at most $f(n)$ such steps. The analysis of one of the convexification schemes would yield $O(n^2)$.

Combinatorial Rigidity and Pseudo triangulations. A 1DOF monotone mechanism obtained from a pseudo-triangulation is an abstraction and a canonical representation of one of the many basic solutions, inducing monotone infinitesimal motions, that the linear programming approach of [12] would find for a certain position of the polygon in its configuration space.

We characterize pseudo-triangulations in several 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 [36] or [17]). 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 [12] based on LP duality and Maxwell's theorem.

2 Definitions and Preliminaries

References. For rigidity theory terminology and basic results, we refer the reader to [30], [36], [37] and [17]. 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 [27] and Maxwell's Theorem are to be found there. For oriented matroids, see [8], although we won't need more than the circular hyperplane (or local) sequences of [9] and [16] (see also [33]).

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 "counterclockwise" as *ccw* and "one-degree-of-freedom mechanism" as *1DOF mechanism*.

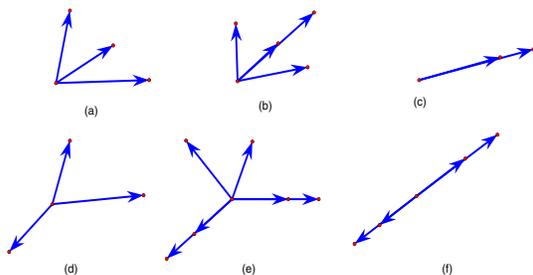


Figure 1: (a), (b) and (c) Acyclic and (d), (e) and (f) cyclic sets of vectors.

Acyclic sets of vectors and edges. A set of vectors in R^2 (with a common origin) is *acyclic* if it is strictly contained in a half-plane, and *cyclic* otherwise. The terminology comes from oriented matroid theory (see [8]), an approach we won't explain in this abstract, but which provided the guiding line in our search for

combinatorial properties of rigid frameworks. More precisely, a set of acyclic vectors has no linear combination with positive, not all zero coefficients that sums them to zero, while in the cyclic case there is always one. Collinearities may occur, see the examples in Fig. 1. The acyclic case of several collinear vectors but in the same direction, and the rest in acyclic relation with them will occur in our algorithm at the beginning of each step. A special cyclic case, which will occur at the end of each step of our convexification algorithm, will have some vectors collinear and in opposite directions, and all the others contained in a half-plane.

An *acyclic set of edges* is a set of segments with endpoints in a finite set of planar points such that the vectors around each vertex, taken in the direction of the adjacent segments and directed away from the vertex, form an acyclic set. See Fig. 2.

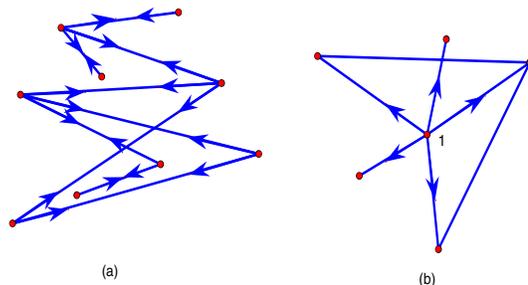


Figure 2: (a) An acyclic set of edges. Around each vertex, the vectors in the directions of the adjacent edges are depicted. (b) A set of edges which is cyclic, due to vertex 1.

Polygons, pseudo-triangles and pseudo k -gons. A *polygon* on the ordered set of points $\mathcal{P} = \{p_1, \dots, p_n\}$ is obtained by joining pairs of consecutive points with line segments (edges of the polygon). It is *simple* if non-adjacent edges do not meet. In this case, there is a well-defined and connected interior and exterior of the polygon. We will assume that the vertices are labeled in ccw order, i.e. such that the interior lies to the left when the boundary of the polygon is traversed in that order.

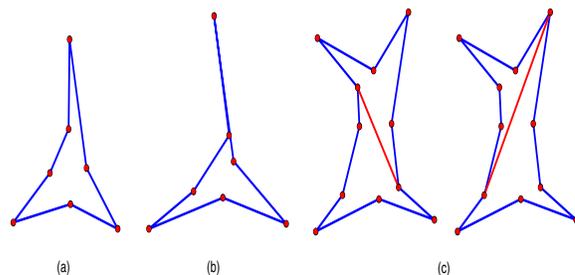


Figure 3: (a) A pseudo-triangle. (b) A semi-simple pseudo-triangle. (c) A pseudo 4-gon and its two possible minimum pseudo-triangulations.

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. We introduce *semi-simple pseudo-triangles* as a special case which allows for some degeneracies: some of the inner convex angles may be zero, but none of the inner reflex angles should be π or 2π . More generally, if we focus on the convex vertices of a simple polygon and on the inner convex chains between them, we will refer to the polygon as being a *pseudo k -gon* if it has exactly k convex vertices. See Fig. 3(c) for an example. *Semi-simple pseudo k -gons* allow simple types of degeneracies: some of the k convex angles may be zero, but none of the reflex angles is π or 2π .

Note. A complete treatment of the general concept of semi-simple polygons as the limit case for simple polygons needs more technical apparatus and will not be covered in this paper. See Fig. 4 for examples and hints to some of the complexities involved in giving a complete definition.

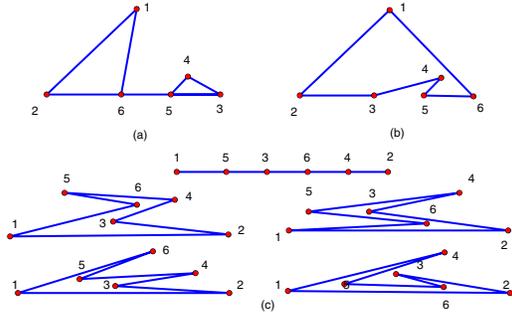


Figure 4: (a) A semi-simple and (b) a simple polygon on the same set of points, and (c) a semi-simple collinear polygon and several possible interpretations (and perturbations) of it. The main question is: which way could an opening motion go?

For the main problem treated in this paper, and if we start with a simple polygon, then the only non simple polygons appearing along the way and which are needed to make our approach work will be semi-simple k -gons.

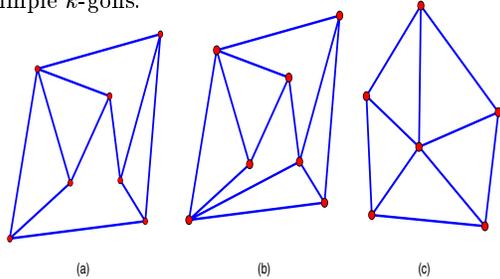


Figure 5: (a) A minimum pseudo-triangulation. (b) A non-minimum pseudo-triangulation which contains a minimum one. (c) A non-minimum pseudo triangulation which does not contain a minimum pseudo triangulation.

Pseudo Triangulations. A pseudo-triangulation of a point set \mathcal{P} is a partitioning of the convex hull of \mathcal{P} into pseudo triangles such that every point appears

as the vertex of at least one pseudo triangular face. We can view it as a collection of edge segments with this property. Equivalently, it is a special embedding of a special planar graph such that the outer face is convex and all internal faces are pseudo triangles. A *minimum pseudo triangulation* is one which has the least number of edges among all pseudo triangulations of the same point set. See Fig. 5.

Pseudo k -gons can also be further subdivided into smaller pseudo l -gons, $l < k$ by the addition of *inner diagonals*. Call that a *pseudo subdivision*. E.g. a pseudo 4-gon can be partitioned into two (or more) pseudo triangles, see Fig.3(c).

Frameworks. A *framework* (G, W) is a graph $G = (V, E)$, $|V| = n$, together with a set of positive weights (lengths) $W = \{l_e | e \in E\}$. A *realization* of (G, W) on a set of points $\mathcal{P} = \{p_1, \dots, p_n\}$ is a mapping of vertices to points and edges to line segments (i.e. an embedding of G) so that the length of the segment corresponding to edge e is l_e . The set of all possible realizations of a framework is called its *configuration space*. As usual, we factor out translations and rotations. The configuration space may be empty, disconnected and in general has a complicated topological structure. The actual values of the edge lengths are not relevant to our discussion, hence by abuse of terminology, from now on we will refer to a realization (G, \mathcal{P}) as a framework. A *motion* or *reconfiguration* of the framework is a curve (one dimensional trajectory) in configuration space going through the point giving the framework realization.

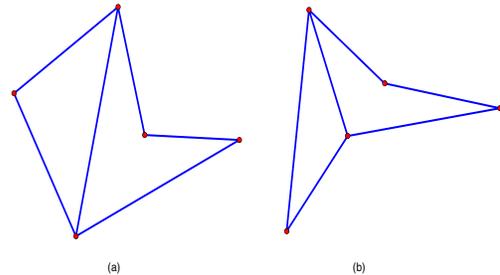


Figure 6: The same graph embedded as two combinatorially distinct frameworks, which are 1DOF mechanisms: (a) is monotone, (b) is not.

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 of the set of points \mathcal{P} . Since in this paper we work only with special types of frameworks, we do not give here the general definition. In our particular case, the information M retained from the embedding will be, for each vertex, the signed circular sequence in which a *directed line* rotating ccw encounters the adjacent edge vectors. 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* of Goodman and Pollack [16] (see also [9] and [33]) and retains (partial) oriented matroid structure from the underlying set of

points P . In particular, we can read off from this information the planar nature of an embedding of the framework and the cyclic or acyclic nature of the edge vectors at each vertex.

A framework is *planar* if its underlying graph is a plane graph (i.e. a planar graph plus the information about its face structure), and is embedded in a manner compatible with the plane graph structure (i.e. all faces are embedded as simple polygons). It is a *pseudo-triangulation* if all internal faces are pseudo-triangles and the outer face is convex. A framework is *acyclic* if all the edge vectors around each vertex are acyclic. We also allow for the special situation (to the best of our knowledge, not dealt with in the literature on planar graph embeddings) when the faces of the embedded framework are semi-simple pseudo k -gons (in particular semi-simple pseudo-triangles). For the careful reader who already noticed that some care must be exercised with the definition of what it means for the outside face to be a semi-simple pseudo k -gon, we rush to add that it can be done in a natural way. First we have to show that the outer face is also some sort of pseudo k -gon. If the outer face is the convex hull of the points, when we “look from the outside” (i.e. from inside the outer face) it has no convex vertex, just a single reflex chain, hence it is an outer pseudo 0-gon. If it has a convex vertex (and hence one reflex chain) it is an outer pseudo 1-gon, etc. Notice that the outer pseudo 3-gon looks quite different from the usual pseudo triangle: its vertices are outside, not inside the convex hull of three convex vertices. For example, the outer face of the framework in Fig. 6(a) is an outer pseudo 1-gon, and for (b) is an outer 2-gon. In general, the outer face will be the only one with this special type of pseudo k -gon. Once we fixed this detail, semi-simplicity is then extended in the obvious way.

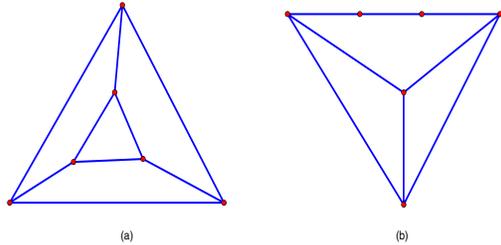


Figure 7: (a) A generically rigid graph in an infinitesimally flexible embedding. (b) A rigid framework with a non-generically rigid underlying graph.

Two 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 lines through the edges around each vertex. This just means that they have the same underlying combinatorial framework (and the embeddings have the same orientation, but we’ll skip this detail here). In particular, we define equivalent pseudo triangulations. The class of all combinatorially equivalent pseudo triangulations

is called a *combinatorial pseudo triangulation*. The underlying combinatorial framework captures the information (and can be used to represent) the combinatorial pseudo triangulation.

A crucial idea in our convexification algorithm will be to use mechanisms obtained from pseudo triangulations, and to reconfigure them continuously as long as the combinatorial pseudo triangulation does not change.

Rigidity, Generic Rigidity, Infinitesimal Rigidity. A framework is (locally) *rigid* if its vertices cannot be moved continuously 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 $\langle \cdot, \cdot \rangle$ is the dot product. Otherwise it is infinitesimally (or first-order) rigid. A graph is *generically rigid* if it is rigid for all embeddings on *generic* sets of points (see the rigidity theory references for precise definitions). Certain embeddings of generically rigid graphs may be infinitesimally flexible, or even flexible. See Fig. 7. Notice that these frameworks are cyclic. A graph is *minimally rigid* if it is rigid and removing some edge creates a graph which is no longer rigid.

Laman’s theorem provides a combinatorial characterization for minimally rigid graphs on generic embeddings: these are graphs 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. A step of type 1 involves adding a vertex, joined by two new adjacent edges to two previously constructed vertices. A step of type 2 adds a new vertex and three adjacent edges to three old vertices such that at least two are joined by an edge, and then drops one of the existing edges among these three old vertices. The reader unfamiliar with these constructions may recognize, in the simplest applications of the type 1 steps which preserve planarity, basic ways of producing triangulations.

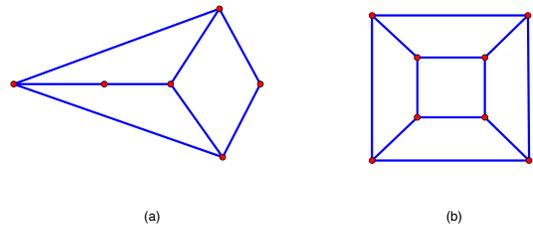


Figure 8: (a) A 1DOF mechanism (Peaucellier’s linkage). (b) A framework with an underlying generically minimally flexible graph, and with a rigid (but not infinitesimally rigid) embedding.

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 (after factorization to translations and rotations). 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.8. Notice that these examples are planar but not acyclic.

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.6 for examples. E.g the Peaucellier linkage in Fig.8 is not monotone, neither is it acyclic.

3 Rigidity of Pseudo-Triangulations

From now on we are interested only in *minimum pseudo triangulations* and show that they have a wealth of combinatorial and rigidity theoretical properties.

Theorem 3.1 (Characterization of minimum 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. G is a minimum pseudo-triangulation.
2. The edges E of G form an acyclic and planar set of segments, and E is maximal with this property (of being both planar and acyclic).
3. G is an acyclic pseudo triangulation of the convex hull of \mathcal{P} .
4. (pseudo triangle Laman count) *The faces of G are pseudo-triangles and the number of edges is $2n - 3$.*
5. (planar acyclic Laman count) *The set of edges E is planar, acyclic and has $2n - 3$ elements.*
6. (planar acyclic Henneberg 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 an acyclic pseudo triangulation). Connect in one of the two ways (see Fig. 9):

(a) Type 1: (degree 2) *Join the vertex with 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.*

(b) Type 2: (degree 3) *Add two tangents as before. Then choose an edge on the convex chain between the two tangent points, remove it. This creates a pseudo 4-gon. Re-pseudo triangulate by adding the unique bitangent different from the one just removed.*

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

For the proofs, we will need a series of basic definitions and facts, which we present in a sketchy manner. Given a point outside a convex hull, a *tangent* from the point to the hull is a line segment containing all the hull vertices on one side and touching it at a vertex. Given two convex hulls, a *bitangent* is a line segment touching each hull in one point and whose supporting line does not separate the vertices on the same hull.

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 (convex hull of the) three inner convex chains.
3. Given a pseudo 4-gon, there exist exactly two ways of adding a bitangent between (the convex hulls of) two inner convex chains. Each one induces a partitioning of the pseudo 4-gon into two pseudo triangles.
4. (*Flips in pseudo triangulations*) Two adjacent pseudo triangles can be flipped: the unique common edge is deleted and replaced with another one (for which there is a unique choice) to obtain again two adjacent pseudo triangles (see Fig.3(c)).

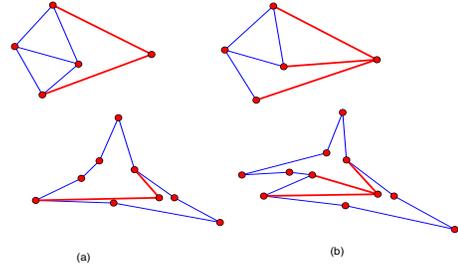


Figure 9: Henneberg steps. (a) type 1 and (b) type 2. Top level, when the new vertex is added on the outside face, bottom level, when it is added inside a pseudo triangular face.

Proof: $1 \rightarrow 2$ This is the most tedious to prove, so we give only a short sketch here. The proof is by contradiction. Assume G contains an acyclic vertex A . Then A is internal (not on the convex hull). Using the above properties, we argue that there is a sequence of deletions of edges, starting with an edge adjacent to A , and re-pseudo-triangulations of the larger faces thus obtained, which contains fewer edges.

$2 \rightarrow 1$ Assume G is maximally planar and acyclic. We prove by contradiction that if G is not a pseudo triangulation, then we can add edges in an acyclic and

planar fashion, thus contradicting maximality. If G is not connected, add bitangents between the convex hulls of different components. If it does not contain its convex hull edges, add them. These operations preserve planarity and acyclicity, hence by maximality we can assume G is both connected and contains the convex hull edges. Similarly, we may argue that its faces are semi-simple polygons. If they are not pseudo triangles, we can always add internal bitangents.

2 \rightarrow 3 follows immediately now, and the reverse from an adaptation of the proof of the first implication.

3 \rightarrow 2 is straightforward.

3 \rightarrow 5 This is one of the interesting parts. We present here a proof based on a *continuous motion* argument. Move the points continuously from the original position to convex position. Changes happen at discrete steps, when three points on the same face become collinear. It is easy to show how to locally patch the pseudo triangulation at each event without increasing the number of edges. When all the points reach convex position, the pseudo triangulation becomes a triangulation of a convex set, which has exactly $2n - 3$ edges.

5 \rightarrow 4 follows from 1 and 2.

The fact that any subset of k vertices induces at most $2k - 3$ edges follows easily, since acyclicity and planarity are hereditary properties (hold on subsets).

5 and 4 \rightarrow 6 We work out the construction in reverse. Because of the edge count, a simple counting argument shows that there must exist at least one vertex of degree strictly less than 4. If there exists a vertex of degree 2, its two adjacent edges are tangent to the face obtained by removing them, because of acyclicity. For a vertex of degree 3, the two extreme edges adjacent to it must be tangents (because of acyclicity). The face obtained by removing the third edge is a pseudo 4 gon (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.

6 \rightarrow 3, 5 and 4 are straightforward: at each step the number of vertices increases by 1 and the number of edges by 2.

□

To simplify the terminology, in the rest of this paper we will refer to a *minimum* or *acyclic* pseudo triangulation as simply a *pseudo-triangulation*.

Theorem 3.2 (Algorithms) *Any acyclic set of edges can be extended to a pseudo-triangulation by arbitrarily adding edges while preserving acyclicity. A canonical extension can be constructed deterministically in $O(n \log n)$ time. If the set of points contains repetitions or collinearities, the resulting faces may be semi-simple pseudo triangles.*

Note that 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. To illustrate that there is not a unique way for constructing them (because just simply adding edges at random while preserving acyclicity would work), we sketch here two possible constructions for pseudo triangulations of polygons which run in $O(n^2)$ time. We will use the second one in the complexity analysis. More efficient constructions (*canonical greedy*), running in $O(n \log n)$ can be obtained via an adaptation of Pocchiola and Vegter's greedy flip algorithm [29]. We wish to thank Michel Pocchiola for pointing this out.

Incremental algorithm. 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.

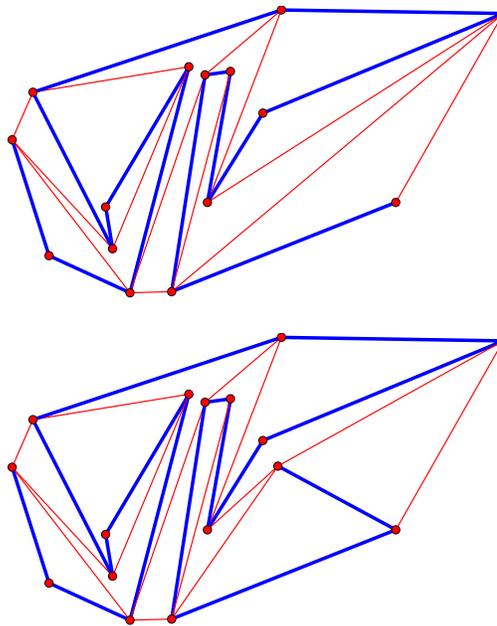


Figure 10: A typical step of the incremental algorithm for computing the pseudo-triangulation of a polygon.

Recursive algorithm. Compute the convex hull of the polygon. The edges of the convex hull which are not edges of the polygon are subdividing the polygon into pockets. Imagine removing the pockets: what is left is a convex polygon. Triangulate it. Then put

back the pockets and *bend* the diagonals of the triangulation along shortest (geodesic) paths inside the polygon.

The process can now be repeated for each pocket. But the convex hulls of pocket subpolygons may intersect (not along polygon edges, but along some of the added hull edges). Again, we will *bend* the added convex hull edges to get *geodesic convex hulls*, then recurse inside each pocket.

The construction has the added advantage of being able to count the number $n - 3$ of added edges via a simple *charging scheme*, whereas an edge is charged to either a geodesic hull of some subpolygon or to a shortest path between two vertices of such a subpolygon. We omit the details here.

We associate a *weight function* to the pseudo triangulation constructed by this algorithm: $f(n)$ = number of bends in all the shortest paths and geodesic hulls of this construction. Since there are linearly many shortest paths and geodesics accounted for (we consider maximal paths only, not subpaths), and each can have at most linearly many bends, $f(n) = \mathcal{O}(n^2)$. When the polygon is convex $f(n) = \mathcal{O}(n)$. We will use this function for the complexity analysis.

Theorem 3.3 (Rigidity property of pseudo triangulations) *Pseudo-triangulations are infinitesimally rigid (and hence rigid), and minimally so (removing an edge the property no longer holds).*

The proof is an adaptation, using the properties of pseudo triangulations, of known proofs of Laman’s theorem for generic rigidity of graphs with the hereditary $(2n - 3)$ -property (every subset of k vertices spans at most $2k - 3$ edges, and there are $2n - 3$ edges in total).

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 [12] (subsequently referred to as the “CDR proof”), 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 the CDR proof 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 this does not

change, the motion continues. This step can be made more precise by an argument as in the CDR proof (based on integration of the resulting vector field), and produces the desired trajectory in configuration space.

To prove the existence of the monotone infinitesimal motion, we also have to adapt slightly the argument in the CDR proof, using LP duality, Maxwell’s Theorem lifting and the mountain-valley argument of [36]. Add all possible diagonals (“struts”) and planarize the graph. By LP duality it suffices to show that there exists no self-stress which is positive on the struts. For the sake of a contradiction, assume there exists such a self-stress and use Maxwell Theorem to lift the picture in 3d to a polyhedral surface. Then 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 can only be along the edges of the pseudo triangulation. This contradicts the acyclicity of the pseudo-triangulation. See Fig. 11, where M is the vertex where the maximum is attained. Just like in the CDR proof, the argument can be easily extended to the case when the maximum is attained on an edge or face, and acyclicity guarantees that we obtain a contradiction.

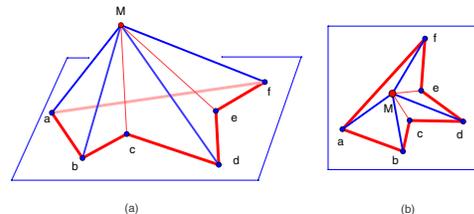


Figure 11: The mountain/valley argument: (a) Cutting just below the vertex of maximum z -coordinate, we get the image in (b).

Our case needs a few more details added to the basic structure of CDR proof. 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 on the interior points. 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. A simple extension of the argument covers the convex hull vertices and semi-simple pseudo triangles. We defer these details 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 these 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 in 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 at alignment events) *When two edges align, one of the following cases two 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 apply 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 to obtain a pseudo triangulation with a semi simple face.*

An example is depicted in Fig. 12.

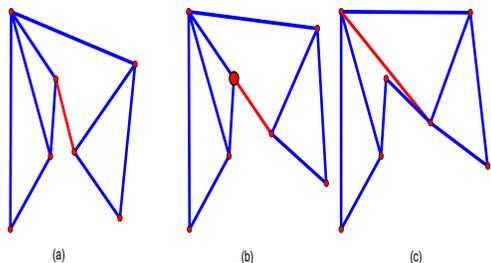


Figure 12: Patching the pseudo triangulation by a local flip when two bars (not both polygon edges) align.

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 argument would work. 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 (see Fig.12).

Another problem occurs 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 unproblematic, and the full description is deferred to the full paper.

Theorem 5.2 Termination and complexity analysis *The convexification of a polygon terminates in finitely many steps, which is at most $\mathcal{O}(n^2)$ steps if we use the weight function based on shortest paths and geodesic hull as invariant.*

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.

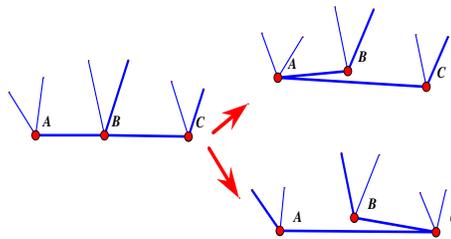


Figure 13: When the two aligning edges do not belong to the polygon, there are two possible diagonal flips. A flip reduces the number of bends in geodesic paths using the vertex involved in the alignment.

A more careful accounting can bring down the number of steps to $\mathcal{O}(n^2)$. We use the weight function introduced for the recursive algorithm in section 3 to keep track of the number of bends in the shortest paths and geodesic hulls. The key observation is that a local flip decreases the number of bends by at least one (or more, depending on how many geodesic paths go through the vertex where the event happened). The special events (when two edges of the polygon align) are only linearly many, and they may not increase the weight function by more than $\mathcal{O}(n)$ each (or: recompute a recursive pseudo triangulation on fewer vertices at that point). The final value of the weight function is at most linear. Hence the number of steps is at most $\mathcal{O}(n^2)$.

Theorem 5.3 Main Result: Convexification of Planar Chains with Monotone Motions *Every planar polygon can be convexified with at most $\mathcal{O}(n^2)$ motions. Each motion is induced by a 1DOF monotone mechanism constructed from a pseudo triangulation with a hull edge removed, which is moved until two of its 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 work space. A first pseudo triangulation can be computed efficiently ($\mathcal{O}(n \log n)$ or $\mathcal{O}(n^2)$) by several algorithms 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 use of pseudo triangulations, whose main combinatorial and rigidity theoretic properties have been described. But still many more await attention, such as: study the graph whose vertices correspond to pseudo triangulations and whose edges correspond to flips in pseudo 4-gons. What is its diameter? How many pseudo triangulations of a point set or polygon are there? Is there any interesting polytope whose 1-skeleton is the graph of pseudo triangulations (as there is for regular triangulations [15])? Since pseudo triangulations can be naturally defined in the coordinate-free context of oriented matroids, what combinatorial properties would distinguish them (as a collection) from the ones realizable in the euclidian plane? Is there a concept of a *regular pseudo triangulation*, as it is for triangulations?

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. 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 degenerate configurations.

All the pictures in this paper have been produced using the software package Cinderella [31], which supports motions of 1DOF mechanisms and even animates them. But not all “pseudo-triangulation-minus-hull-edge” mechanisms admit a Cinderella construction. It would be interesting to classify pseudo triangulations in a complexity hierarchy, based on the extra primitives that should be included in Cinderella to simulate them (i.e. added to basic ruler and compass constructions). This is also related to the problem of computing (in the real RAM model) the coordinates of the points realizing a configuration of a pseudo triangulation mechanism at one edge alignment moment, given the coordinates of the points at the previous event. It probably cannot be done better than using standard numerical approximation techniques.

A web page containing Cinderella animations and other graphical and 3d illustrations of our approach can be found at the author's url, <http://cs.smith.edu/streinu>. Thanks to my students Beenish Chaudry, Victoria Manfredi, Christine Rice and Elif Tosun for their help.

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, J.E. Goodman, Anna Lubiw, R. Pollack, E. Welzl and Sue Whitesides) and to the many participants (too many to list here, but see [7], [6]) for enthusiastic conversations on this and related problems.

I am particularly indebted to Günter Rote, whose generous sharing of his own ideas (on the use of monotone motions and the linear programming approach to this problem) provided the inspiration for all my subsequent work.

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 mathematics literature and for stimulating conversations.

References

- [1] B. Aronov, J.E. Goodman and R. Pollack. *Convexifying a Planar polygon in R^3* . manuscript, 1999.
- [2] P. Agarwal, J. Basch, L. Guibas, J. Hershberger and L. Zhang. *Deformable Free Space Tilings for Kinetic Collision Detection*, Proc. 4th Int. Workshop on Alg. Found. Robotics (WAFR), Dartmouth College, March 2000,
- [3] J. Basch, J. Erickson, L. Guibas, J. Hershberger and L. Zhang. *Kinetic Collision Detection Between Two Simple Polygons*, Proc. 10th ACM-SIAM Symp. Discr. Algorithms (SODA), 1999, pp.102-111.
- [4] S. Basu, R. Pollack and M.-F. Roy. *Computing Roadmaps of Semi-algebraic Sets on a Variety*, JAMS, 13, pp. 55-82, 2000.
- [5] S. Basu, R. Pollack and M.-F. Roy. *Algorithms in Real Algebraic Geometry*, book, in preparation.
- [6] 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.
- [7] 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.
- [8] A. Björner, M. Las Vergnas, B. Sturmfels, N. White and G. Ziegler. *Oriented Matroids*, Cambridge University Press, 1993
- [9] J. Bokowski. *Oriented Matroids*, in P.M. Gruber and J.M. Wills, *Handbook of Convex Geometry*, North Holland, 1993.
- [10] J. Canny. *Complexity of Robot Motion Planning*, MIT Press, Cambridge, MA, 1988.
- [11] R. Cocan and J. O'Rourke. *Polygonal Chains Cannot Lock in 4d*, Proc. 11th Canad. Conf. Comput. Geometry, Vancouver, 1999, pp. 5-8.
- [12] R. Connelly, E. Demaine and G. Rote. *Straightening Polygonal Arcs and Convexifying Polygonal Cycles*, this proceedings, 2000
- [13] H. Crapo and W. Whiteley, *Plane self-stresses and projected polyhedra I: The basic pattern*, Structural Topology 20, 1993, pp. 55-77.
- [14] P. Erdős. Problem number 3763. Am. Math. Monthly, 46:627, 1935.
- [15] I. Gelfand, M. Kapranov and A. Zelevinsky. *Newton polytopes of principal A-determinants*, Soviet Math. Doklady 40, 1990, pp. 278-281.
- [16] 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
- [17] J. Graver, B. Servatius and H. Servatius. *Combinatorial Rigidity*, Amer. Math. Soc., Graduate Studies in Mathematics vol.2, 1993.
- [18] L. Han and N. Amato *A Kinematics-Based Probabilistic Roadmap Method for Closed Chain Systems*. Proc. 4th Int. Workshop on Alg. Found. Robotics (WAFR), Dartmouth College, March 2000.
- [19] 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.
- [20] D. Jordan and M. Steiner. *Configuration Spaces of Mechanical Linkages*, Discrete Comput. Geometry 22, pp. 297-315, 1999.
- [21] D. Joseph and W.H. Platinga. *On the complexity of reachability and motion planning questions*, Proc. Symp. Comp. Geom., pp. 62-66, 1985.
- [22] 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.
- [23] M. Kapovich and J. Millson. *Universality Theorems for configurations spaces of planar linkages*, preprint, 1998.
- [24] 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.
- [25] R. Kirby. *Problems in Low-Dimensional Topology*. <http://www.math.berkeley.edu/~kirby>, 1995.
- [26] W.J. Lenhart and Sue H. Whitesides. *Reconfiguring Closed Polygonal Chains in Euclidian d-space*. Discr. Comp. Geometry 13:123-140, 1995.
- [27] L. Lovasz and Y. Yemini. *On generic rigidity in the plane*, SIAM J. Algebraic and Discrete Methods, 3, no. 1, 91-98.
- [28] A. Lubiw and S. Whitesides (org.), McGill University Workshop on Folding and Unfolding, Barbados, 1998, informal proceedings.
- [29] M. Pocchiola and G. Vegter *Topologically Sweeping Visibility Complexes via Pseudo-Triangulations*, Discrete Comput. Geometry 16(4), 419-453, Dec. 1996.
- [30] A. Recki. *Matroid Theory and Its Applications*, Springer Verlag, 1989.
- [31] J. Richter-Gebert and U. Kortenkamp. *Cinderella, an interactive software package for Geometry*, Springer Verlag, 1999.
- [32] 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.
- [33] I. Streinu. *Clusters of Stars*, Proc. ACM Symp. Comp. Geometry, Nice, 1997, pp. 439-441.
- [34] G. Toussaint. *The Erdős-Nagy Theorem and its Ramifications*, Proc. Canad. Conf. Comp. Geom., 1999.
- [35] W. Whiteley. *Motions and Stresses of Projected Polyhedra*, Structural Topology 7, pp. 13-38, 1982. Amer. Math. Soc. 1996.
- [36] W. Whiteley. *Some Matroids from Discrete Applied Geometry*, Contemporary Mathematics, vol. 197, pp. 171-311, Amer. Math. Soc. 1996.
- [37] W. Whiteley. *Rigidity and Scene Analysis*, in *Handbook of Discrete and Computational Geometry*, (J.E. Goodman, J. O'Rourke, eds.), 1997, pp. 893-916.
- [38] S.H. Whitesides. *Algorithmic issues in the geometry of planar linkage movement*, Austral. Comp. J., Spec. Issue on Algorithms, 24(2):42-50, 1992.