Recognizing Planar Laman Graphs

Jonathan Rollin[1], Lena Schlipf[1], and André Schulz[1]

1 **FernUniversität in Hagen**
`{jonathan.rollin | lena.schlipf | andre.schulz}@fernuni-hagen.de`

Abstract

Laman graphs are the minimally rigid graphs in the plane. We present two algorithms for recognizing planar Laman graphs. A simple algorithm with running time $O(n^{3/2})$ and another one with running time $O(n \log^3 n)$ based on latest planar network flow algorithms. Both improve upon the previously fastest algorithm for general graphs by Gabow and Westermann [*Algorithmica*, 7(5-6):465–497, 1992] with running time $O(n\sqrt{n \log n})$.

1 Introduction

Let $G = (V, E)$ be a graph with n vertices. The graph G is called a *Laman graph* if it has $2n - 3$ edges and every subset $V' \subseteq V$ induces a subgraph with no more than $2|V'| - 3$ edges. A *bar-joint framework* is a physical structure made from fixed-length bars that are linked by universal joints (allowing 360° rotations) at their endpoints. A bar-joint framework is *flexible* if it has a motion other than a global rotation or translation. A nonflexible framework is called *rigid*. Moreover it is called *minimally rigid*, if it is rigid, but it becomes flexible after removing any bar. Interestingly, in 2d a bar-joint framework (in a generic configuration) is minimally rigid, if and only if its underlying graph is a Laman graph.

Various characterizations of Laman graphs are known [9, 14, 15]. The class of *plane* Laman graphs provides even more structure [8, 12]. Of particular interest for our result is the following geometric characterization: A geometric graph is a *pointed pseudotriangulation* (PPT) if each inner face contains exactly three angles less than π, called *small*, and every vertex is incident to an angle larger than π, called *big* [19]. Streinu [20] proved that PPTs are Laman graphs. Moreover, Haas et al. [8] showed that every planar Laman graph can be realized as a PPT.

The concept of pointed pseudotriangulations can be transferred to plane (abstract) graphs. In fact, this was an intermediate step in the proof by Haas et al. A *combinatorial pointed pseudotriangulation* (CPPT) is a plane graph (with $2n - 3$ edges) with an assignment of the labels "small"/"big" to the angles satisfying the properties of a PPT. Not every CPPT can be stretched to a PPT, but every plane Laman graph admits a CPPT assignment and

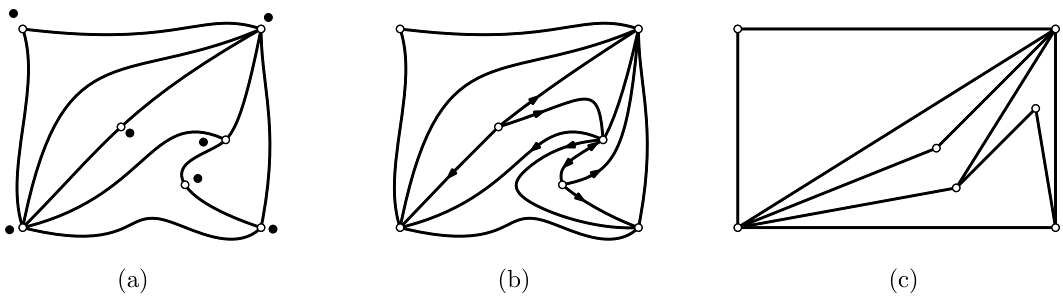

Figure 1 A CPPT with big angles drawn marked by solid circles (a), the derived directed graph $\vec{G}$ (b), the stretched pointed pseudotriangulation (c).

35th European Workshop on Computational Geometry, Utrecht, The Netherlands, March 18–20, 2019.
This is an extended abstract of a presentation given at EuroCG'19. It has been made public for the benefit of the community and should be considered a preprint rather than a formally reviewed paper. Thus, this work is expected to appear eventually in more final form at a conference with formal proceedings and/or in a journal.

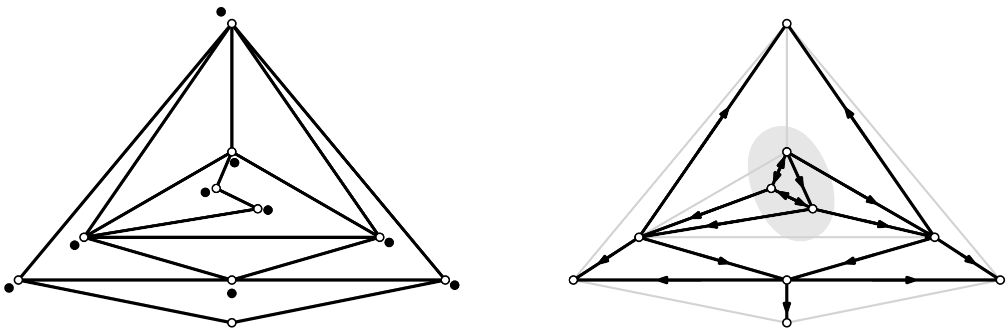

Figure 2 A CPPT that is not stretchable (a) and the derived directed graph $\vec{G}$ where the highlighted vertices do not have 3 disjoint paths to the outer face (b).

each CPPT which is a Laman graph is stretchable. Haas et al. [8] provide an algorithm with running time $O(n^{3/2})$ that finds a CPPT assignment for a planar Laman graph.

In order to recognize plane Laman graphs it is sufficient to check whether a given CPPT is stretchable. We provide such algorithms based on the following characterization of stretchability by Haas et al. by means of connectivity.

▶ **Lemma 1.1** ([8]). *For a CPPT G a directed plane graph $\vec{G}$, with $V(\vec{G}) = V(G)$, can be computed in linear time such that G is stretchable if and only if for each interior vertex $u \in \vec{G}$ there are 3 vertex disjoint directed paths from u to distinct vertices on the boundary of $\vec{G}$.*

The condition of Lemma 1.1 seems to be an interesting property on its own, since it can be understood as a form of "directed 3-connectivity".

1.1 Our contribution

Consider a positive integer k, a directed graph G, and disjoint sets S, $T \subseteq V(G)$. We call S *k-connected to T* if for each vertex $s \in S$ there are k directed paths from s to T pairwise having only vertex s in common.

▶ **Theorem 1.2.** *For each fixed $k \geq 1$ there is an algorithm deciding for a directed planar graph G and a partition $V(G) = S \cup T$ whether S is k-connected to T in $O(n^{3/2})$ time.*

We present the simple algorithm for Theorem 1.2 in Section 2. To check the Laman property for a plane graph G we use the algorithms of Haas et al. [8] to find a CPPT assignment and the directed plane graph $\vec{G}$ from Lemma 1.1, and then the algorithm from Theorem 1.2 to decide whether the set of interior vertices of $\vec{G}$ is 3-connected to the set of boundary vertices. This decides whether G is a plane Laman graph by Theorem 1.2 and Lemma 1.1 and has running time $O(n^{3/2})$.

A faster algorithm is obtained as follows. To search for a CPPT assignment we use an algorithm of Borradaile et al. [3]. This algorithm computes a maximum flow between multiple sources and sinks in $O(n \log^3 n)$ time. To check the connectivity condition we first use a construction similar to one of Kaplan and Nussbaum [11]. Then an algorithm due to Łącki et al. [13] is used that computes for a single source a maximum flow to each other vertex in $O(n \log^3 n)$ time. Details of this faster algorithm are presented in Section 3.

▶ **Theorem 1.3.** *The recognition problem for planar Laman graphs can be solved in $O(n \log^3 n)$.*

Checking the Laman condition for general graphs can be done in polynomial time. The fastest (but very complicated) algorithm is due to Gabow and Westermann [6] (see also [4]) and needs $O(n\sqrt{n}\log n)$ time. Their algorithm is based on a characterization by means of matroid sums. There is also a very easy pebble-game algorithm that runs in $O(n^2)$ time [15]. For planar graphs Haas et al. [8] give an algorithm that computes a PPT from a Laman graph in time $O(n^{3/2})$. Their algorithm can be turned into a recognition algorithm by checking whether the derived realization is a PPT. If the original graph is not a Laman graph some parts of the drawing collapse. Such a check however requires computations with exponentially large numbers. So it depends on the model of computation if the overall algorithm runs in $O(n^{3/2})$ time. Our combinatorial algorithms avoid these subtleties.

2 Proof of Theorem 1.2

We first give some structural results. The following statement is similar to Menger's theorem.

▶ **Lemma 2.1.** *Let $k \geq 0$, G be a directed (not necessarily planar) graph, and S, $T \subseteq V(G)$ be disjoint with $|T| \geq k$. Then S is k-connected to T if and only if for each $s \in S$ and $A \subseteq V(G) \setminus \{s\}$ with $|A| = k-1$ there is a directed path from s to T not using the vertices in A.*

▶ **Lemma 2.2.** *Let G be a directed graph and let S, T, $T' \subseteq V(G)$ be disjoint. If S is k-connected to $T \cup T'$ and T' is k-connected to T, then S is k-connected to T.*

Proof. Let $s \in S$ and fix a set $A \subseteq V(G) \setminus \{s\}$ of size $k-1$. There is a directed path from s to some vertex $u \in T \cup T'$ not using vertices from A. If $u \in T$ we are done. If $u \in T'$, then there is a directed path from u to T not using vertices from A. In each case there is a directed path from s to T not using vertices from A. So S is k-connected to T by Lemma 2.1. ◀

For a single vertex we can decide in $O(kn)$ time whether it is k-connected to T as follows. A slight modification of a by now standard construction due to Ford and Fulkerson [5] gives a directed graph G' and vertices s', $t' \in V(G')$ such that $s \in V(G)$ is k-connected to T in G if and only if G' admits an s'-t'-flow with value at least k. To check whether G' admits such a flow we use at most k steps of augmentation in Ford–Fulkerson's algorithm. Since each augmentation step needs only linear time we have the following result.

▶ **Lemma 2.3.** *For each $k \in \mathbb{N}$ there is an algorithm deciding for any directed (not necessarily planar) graph G, $s \in V(G)$, and $T \subseteq V(G)$ whether s is k-connected to T in linear time.*

We call $A \subseteq V(G)$ a *separator* if removing A splits G into two (not necessarily connected) subgraphs G_1 and G_2, such that $|V(G_1)|$, $|V(G_2)| \leq \frac{2}{3}|V(G)|$. For every planar graph G a separator with size in $O(\sqrt{n})$ can be found in linear time [16].

Proof of Theorem 1.2. The algorithm works recursively as follows. Let A denote a separator of G of size $O(\sqrt{n})$. Use Lemma 2.3 to check for each $a \in A \cap S$ whether a is k-connected to T in G. Let G_1 and G_2 denote the two subgraphs of G separated by A (each including A). For $i = 1, 2$ let $S_i = (S \cap V(G_i)) \setminus A$ and let $T_i = (T \cap V(G_i)) \cup A$. Apply the algorithm recursively to check whether S_i is k-connected to T_i in G_i, for $i = 1, 2$.

The algorithm indeed checks whether S is k-connected to T in G since either it finds some vertex in $A \cap S$ that is not k-connected to T in G, or it is sufficient to check whether $S \setminus A$ is k-connected to $A \cup T$ by Lemma 2.2. Then it is sufficient to check G_1 and G_2 separately.

The separator can be found in linear time. Then the algorithm from Lemma 2.3 is called $O(\sqrt{n})$ times for each vertex in the separator, each call with time in $O(n)$. So the total time for each step of the recursion and hence for the whole algorithm is $O(n^{3/2})$. ◀

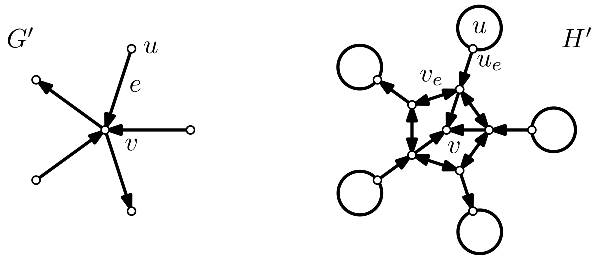

Figure 3 The modification applied to a planar directed graph G'. The modification of Kaplan and Nussbaum does not include the original vertices inside of the new cycles. We need this since all the original vertices except a fixed source are targets in our algorithm.

3 Proof of Theorem 1.3

Here we describe how to use flow algorithms for planar graphs to check whether a plane graph G admits a CPPT assignment and whether a CPPT is stretchable.

The vertex-face incidence graph of a plane graph $G = (V, E)$ with face set F is a planar bipartite graph $H = (V \cup F, E')$ where $(v, f) \in E'$ if and only if $v \in V$ is incident to $f \in F$ in G. A plane graph G admits a CPPT assignment if and only if its vertex-face incidence graph has a subgraph H' where each interior face of G has degree 3 in H', the outer face of G has degree 0 in H', and each vertex of G has degree in H' equal to its degree in G minus 1 [8]. This means that edges of H' correspond to small angles in the CPPT assignment. Via some standard techniques we can use the algorithm of Borradaile et al. [3] (which computes an integer flow) to find such an assignment in $O(n \log^3 n)$ time if it exists.

To check whether a CPPT is stretchable using the algorithm of Łącki et al. [13] for maximum flow we need the following result similar to Lemma 1.1.

▶ **Lemma 3.1.** *For each CPPT G a directed planar graph $\vec{G}$ with $|V(\vec{G})| \leq 7|V(G)|$ can be computed in linear time together with some $s \in V(\vec{G})$ and $T \subseteq V(\vec{G})$ such that G is stretchable if and only if in $\vec{G}$ for each $t \in T$ the value of a maximum s-t-flow is at least* 3.

Having this lemma our algorithm to recognize planar Laman graphs works as described in the introduction. It remains to prove Lemma 3.1. To this end we modify a construction of Kaplan and Nussbaum [11]. Consider a directed graph G and distinct s, $t \in V(G)$. Kaplan and Nussbaum construct a directed planar graph $G_{s,t}$ obtained from G by replacing each $u \in V(G) \setminus \{s, t\}$ by a cycle C_u with vertices $u_1, \ldots, u_d$, where d is the degree of u in G, such that arcs incident to u are replaced by arcs not sharing endpoints while keeping their orientation and the cyclic order around u. The edges of the cycle C_u are oriented in both directions and receive (flow) capacity 1/2. See Figure 3.

▶ **Lemma 3.2** ([11]). *There are k internally vertex disjoint s-t-paths in G if and only if in $G_{s,t}$ the maximum s-t-flow has value at least k.*

Proof of Lemma 3.1. Consider a CPPT G. Let G' denote a directed planar graph given by Lemma 1.1, that is, G is stretchable if and only if in G' the set of interior vertices is 3-connected to the set of boundary vertices. Let H denote the directed planar graph obtained by reversing the direction of each arc in G' and by adding a new vertex s in the outer face of G' connected by arcs sv to all boundary vertices v of G'. Then in G' the set of interior vertices T is 3-connected to the set of boundary vertices if and only if in H there are 3 internally vertex disjoint s-t-paths for each $t \in T$.

We obtain a directed planar graph H' by splitting each vertex in $V(H) \setminus \{s\}$ into a cycle as follows. Replace each arc e in H, directed from $u \neq s$ to v, by arcs $u_e v_e$ and $v_e v$, where u_e, v_e are new vertices (and distinct for all e). Replace each arc e in H, directed from s to v, by arcs sv_e, and $v_e v$, where v_e is a new vertex. Further for each vertex u in H connect the new vertices u_e by a cycle C_u in the cyclic order of the arcs e around u, where the edges are directed in both directions. Finally a (flow) capacity function is defined where all arcs on cycles C_u receive capacity $1/2$ and all other arcs capacity 1. See Figure 3.

This construction corresponds to the graph $H_{s,t}$ constructed by Kaplan and Nussbaum, except that there is not a specific target t and, additionally the original vertices from H are kept inside of the cycles together with their incoming arcs. In particular H' is planar and $V(H) \subseteq V(H')$. Consider some $t \in V(H)$. Note that each s-t-flow in H' does not use vertices from $V(H) \setminus \{s, t\}$, since these vertices do not have outgoing arcs. Hence for each $t \in V(H)$ any s-t-flow in H' corresponds to an s-t-flow in H_E (by contracting t and C_t to a single vertex). By Lemma 3.2 there are 3 internally vertex disjoint s-t-paths in H if and only if in H' the value of a maximum s-t-flow is at least 3.

Clearly $\vec{G}$ can be constructed in linear time and $|V(\vec{G})| \leq |V(G)| + 2|E(G)| + 1 \leq 7|V(G)|$. This shows that $\vec{G} = H'$ together with the set T satisfies the desired conditions. ◀

4 Conclusions and further directions

An obvious direction for future research is to search for faster or simpler algorithms recognizing (planar) Laman graphs.

Our algorithms do not provide any certificate for their correctness. This could be a Henneberg sequence [9] or a decomposition into two acyclic subgraphs [4]. We do not know how to compute either of these faster than using the algorithm of Gabow and Westermann [6].

Finally, it remains to improve the running time for nonplanar graphs. Notice that our approach heavily depends on planarity. However, it is of independent interest to see if the connectivity results for directed graphs can be extended. We can adapt the ideas presented in the first algorithm when the graph has a small separator. The running time becomes linear for graphs with separators of constant size and stays in $O(n^{3/2})$ as long as there are separators of size $O(\sqrt{n})$. Similar variants of connectivity were studied before, for instance the all-pairs reachability [7, 10], all pairs minimum cut [2], or the vertex disjoint path or Menger problem [18]. We are not aware of other related results.

Instead of asking if a graph has a representation as a pointed pseudotriangulation one can ask for other representations such as general pseudotriangulations [17, 19] or straight-line triangle representations [1]. For the latter no polynomial time algorithm is known.

References

1 Nieke Aerts and Stefan Felsner. Straight line triangle representations. *Discrete Comput. Geom.*, 57(2):257–280, 2017.

2 Glencora Borradaile, David Eppstein, Amir Nayyeri, and Christian Wulff-Nilsen. All-pairs minimum cuts in near-linear time for surface-embedded graphs. In *32nd International Symposium on Computational Geometry*, volume 51 of *LIPIcs. Leibniz Int. Proc. Inform.*, pages Art. 22, 16. Schloss Dagstuhl. Leibniz-Zent. Inform., Wadern, 2016.

3 Glencora Borradaile, Philip N. Klein, Shay Mozes, Yahav Nussbaum, and Christian Wulff-Nilsen. Multiple-source multiple-sink maximum flow in directed planar graphs in near-linear time. *SIAM J. Comput.*, 46(4):1280–1303, 2017.

4 Ovidiu Daescu and Anastasia Kurdia. Towards an optimal algorithm for recognizing Laman graphs. *J. Graph Algorithms Appl.*, 13(2):219–232, 2009.

5 L. R. Ford, Jr. and D. R. Fulkerson. *Flows in networks.* Princeton University Press, Princeton, N.J., 1962.

6 Harold N. Gabow and Herbert H. Westermann. Forests, frames, and games: algorithms for matroid sums and applications. *Algorithmica*, 7(5-6):465–497, 1992.

7 Loukas Georgiadis, Daniel Graf, Giuseppe F. Italiano, Nikos Parotsidis, and Przemysław Uznański. All-pairs 2-reachability in $O(n^\omega \log n)$ time. In *44th International Colloquium on Automata, Languages, and Programming*, volume 80 of *LIPIcs. Leibniz Int. Proc. Inform.*, pages Art. No. 74, 14. Schloss Dagstuhl. Leibniz-Zent. Inform., Wadern, 2017.

8 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. *Comput. Geom.*, 31(1-2):31–61, 2005.

9 Lebrecht Henneberg. Die Graphische Statik der Starren Körper. In Felix Klein and Conrad Müller, editors, *Encyklopädie der Mathematischen Wissenschaften mit Einschluss ihrer Anwendungen: Vierter Band: Mechanik*, pages 345–434. Vieweg+Teubner Verlag, Wiesbaden, 1908.

10 Jacob Holm, Eva Rotenberg, and Mikkel Thorup. Planar reachability in linear space and constant time. In *2015 IEEE 56th Annual Symposium on Foundations of Computer Science—FOCS 2015*, pages 370–389. IEEE Computer Soc., Los Alamitos, CA, 2015.

11 Haim Kaplan and Yahav Nussbaum. Maximum flow in directed planar graphs with vertex capacities. *Algorithmica*, 61(1):174–189, 2011.

12 Stephen Kobourov, Torsten Ueckerdt, and Kevin Verbeek. Combinatorial and geometric properties of planar Laman graphs. In *Proceedings of the Twenty-Fourth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1668–1678. SIAM, Philadelphia, PA, 2012.

13 Jakub Łącki, Yahav Nussbaum, Piotr Sankowski, and Christian Wulff-Nilsen. Single source – all sinks max flows in planar digraphs. *2012 IEEE 53rd Annual Symposium on Foundations of Computer Science*, pages 599–608, 2012.

14 Gerard Laman. On graphs and rigidity of plane skeletal structures. *J. Engrg. Math.*, 4:331–340, 1970.

15 Audrey Lee and Ileana Streinu. Pebble game algorithms and sparse graphs. *Discrete Math.*, 308(8):1425–1437, 2008.

16 Richard J. Lipton and Robert E. Tarjan. A separator theorem for planar graphs. *SIAM J. Appl. Math.*, 36(2):177–189, 1979.

17 David Orden, Francisco Santos, Brigitte Servatius, and Herman Servatius. Combinatorial pseudo-triangulations. *Discrete Math.*, 307(3-5):554–566, 2007.

18 Heike Ripphausen-Lipa, Dorothea Wagner, and Karsten Weihe. The vertex-disjoint Menger problem in planar graphs. *SIAM J. Comput.*, 26(2):331–349, 1997.

19 Günter Rote, Francisco Santos, and Ileana Streinu. Pseudo-triangulations—a survey. In *Surveys on discrete and computational geometry*, volume 453 of *Contemp. Math.*, pages 343–410. Amer. Math. Soc., Providence, RI, 2008.

20 Ileana Streinu. Pseudo-triangulations, rigidity and motion planning. *Discrete Comput. Geom.*, 34(4):587–635, 2005.