New results on Pseudo-triangulations with low vertex degree

André Schulz*

Abstract

We show that it is NP-hard to decide if a geometric graph can be extended to a pseudo-triangulation by adding lines and not violating a given degree bound. The equivalent problem for triangulations is known to be NP-hard if we ask for a triangulation with maximum vertex degree 7 [3]. We show that finding a pseudo-triangulations is already NP-complete if we forbid vertex degrees greater than 5. The problem stays NP-complete for pointed pseudo-triangulations. As a second result we show that point sets with uniformly distributed convex layers allow a pseudo-triangulation with low degree and face bound.

1 Introduction

A *pseudo-triangle* is a polygon with exactly three convex corners. A planar partition of a point set into pseudo-triangles is called *pseudo-triangulation*. A pseudo-triangulation is *pointed* if every vertex is incident to an angle greater than π. Pointed pseudo-triangulations are also called *minimum*, because it is not possible to pseudo-triangulate a point set with fewer edges. Many different applications for (pointed) pseudo-triangulations are known (e.g. ray shooting[2], visibility problems[7], collision detection[5]).

It is always possible to construct a pointed pseudo-triangulation with maximal vertex degree 5 in $O(n \log n)$ time [4]. This bound is tight. The degree bound can be used to design efficient data structures to store pseudo-triangulations. The degree of a triangulation can't be bounded by a constant.

Let E be a set of non-crossing edges. We want E to be part of the pseudo-triangulation. This is called a *constrained* pseudo-triangulation. It is always possible to extend non-crossing edges to a pseudo-triangulation. Also for constrained pseudo-triangulations degree bounds are known (e.g. a simple polygon can be pseudo-triangulated with no vertex exceeding a degree of 5) [1]. It is an open question which point sets allow pseudo-triangulations with smaller degree bound. In Section 2 we prove that at least for constrained pseudo-triangulations this question is NP-complete.

Even though for every point set there exists a pseudo-triangulations with degree bound 5, it is not possible to realize small vertex and face degree together. An example which proved that the product of face and vertex degree is in $\Theta(n)$ is given in [4]. In Section 3 we characterize point sets which allow better results in this setting.

*Institut für Informatik, Freie Universität Berlin,Takustraße 9, D-14195 Berlin, Germany, email:`schulza@inf.fu-berlin.de`
Supported by the Deutsche Forschungsgesellschaft (DFG) under grant RO 2338/2-1

2 NP-completeness

Let $V = \{v_1, \dots, v_n\}$ be a set of n vertices in the plane. $G = (V, E)$ denotes a planar geometric graph. If the outer face of G forms a polygon we call it *polygonal*. G may contain internal points, lines and holes. We define the following problems:

Constrained pseudo-triangulation with degree bound k (`k-CPT`)
Input: $G = (V, E)$
Question: Is there a pseudo-triangulation of V with constraint E and maximal vertex degree k?

Constrained Polygon pseudo-triangulation with degree bound k (`k-CPPT`)
Input: $G = (V, E)$, G polygonal
Question: Is there a pseudo-triangulation of the polygon induced by G with constraint E and maximal vertex degree k?

Constrained pointed Polygon pseudo-triangulation with degree bound k (`k-CPPPT`)
Input: $G = (V, E)$, G polygonal
Question: Is there a pointed pseudo-triangulation of the polygon induced by G with constraint E and maximal vertex degree k?

Theorem 1 *For any $k \geq 5$ `k-CPPT` and `(k+1)-CPT` are NP-complete.*

The proof of the Theorem 1 will be given by the following discussion. Clearly all three problems are in NP. The NP-hardness will be shown by reducing `PLANAR-3-SAT` to `k-CPPT` or respectively to `k-CPT`. First we will show the reduction to `5-CPPT`. It is known that `3-SAT` is NP-complete. Lichtenstein [6] proved that it is even NP-complete when the formula ϕ is planar. A formula is planar if it can be represented as a planar graph $G(\phi) = (V_\phi, E_\phi)$. The set V_ϕ is given by the variables and the clauses of ϕ. The pairs of all (negated) variable and its associated clauses define E_ϕ.

The reduction is done by constructing an instance of 5-CPPT for a formula ϕ. The geometric graph G will be generated in such a way, that it can only be pseudo-triangulated with maximum degree 5, if ϕ is satisfiable. Otherwise at least one vertex must have degree 6.

The starting point for the construction of G is an arbitrary (polygonal) embedding of $G(\phi)$. G will be constructed by substituting vertices and edges of $G(\phi)$ by *gadgets*. Edges will be substituted by the WIRE-gadget, for the clauses we need NOT-gadgets and NAND-gadgets. Finally we introduce the SPLIT-gadget which is needed to duplicate truth values. The variables of ϕ doesn't need a separate gadget - they will be decoded as a part of the WIRE-gadget. The gadgets are defined as follows:

WIRE-gadget: Since we forbid vertex degree greater than 5, the WIRE-gadget can be pseudo-triangulated in exactly two ways. We associate these pseudo-triangulations with "true" and "false". Figure 1 shows the gadget. We are free to choose one of the two possible diagonals in the left pseudo-quadrilateral. This choice determines all other diagonals - otherwise we would violate the degree bound. It is possible to

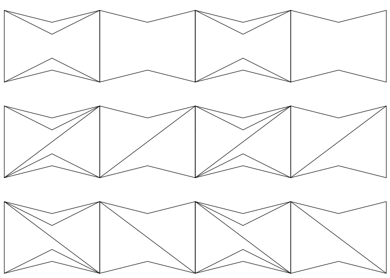
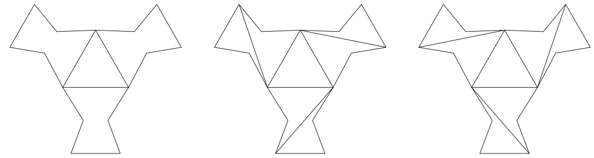


Figure 1: The wire gadget (top most) and the two possible pseudo-triangulations beneath.

simulate turns by bending the wire slightly . The structure and properties of the gadget will stay the same.

SPLIT-gadget: Since we have only one copy of each variable, but different clauses where it could appear, we need to *split* the wire. The gadget has an input and two output interfaces. Like for the wire, we have only two different choices to pseudo-triangulate the gadget. Both ways duplicate the "input signal" and send it to the output part. Figure 2 shows the gadget and its possible pseudo-triangulations.

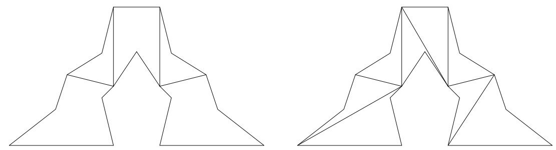
NOT-gadget: The NOT-gadget(Figure 3.a) always comes along with a SPLIT-gadget. It has two input interfaces (at the bottom of the figure) which will receive the same "signal". This will be realized by the SPLIT-gadget. With this assumption there are again



(a) (b) (c)

Figure 2: The SPLIT-gadget (a) and its pseudo-triangulations (b) & (c).

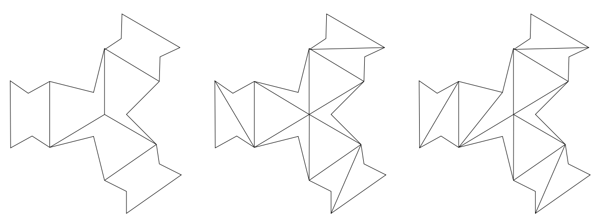
only two pseudo-triangulations with degree at most 5 left. Both will reverse the direction of the diagonals inside the wire. Figure 3.b shows one of the two valid pseudo-triangulation.



(a) (b)

Figure 3: The NOT-gadget (a) and its pseudo-triangulation (b).

NAND-gadget: To simulate the clauses of ϕ we introduce the NAND gadget (Figure 4.a). It has 3 input interfaces. When all diagonals of the input wires are turned right, all 3 diagonals of the NAND-gadget will meet in the central point of the gadget (Figure 4.b). That leads to a vertex with degree 6 and therefore violates the degree condition. If only one input diagonal is flipped we can construct a valid pseudo-triangulation (e.g. see Figure 4.c).



(a) (b) (c)

Figure 4: The NAND-gadget (a) with a forbidden (b) and a valid (c) pseudo-triangulation.

Combining all the arguments for the gadgets it follows that ϕ is satisfiable, if and only if there is a pseudo-triangulation with maximal vertex degree 5. This proves the NP-hardness for 5-CPPT.

The result of Jansen[3] was achieved with a similar reduction. Our approach uses more sophisticated gadgets and the advantages of *pseudo*-triangles. Thus we are able to show NP-completeness for $k = 5$ instead of $k = 7$ like it was done in [3] for triangulations.

To show NP-completeness for `k-CPPT` where $k > 5$ we modify the gadgets slightly. The boundary of each gadget consist of pointed vertices. This allows us to add slightly more bended copies of the concave chains of the gadgets (see Figure 5). Thus the "original" vertex degree will grow and hence the maximum vertex degree of the pseudo-triangulation. Additionally we must increase the degree for the central vertex of the NAND-gadget like it was done in Figure 5.

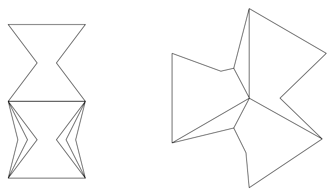


Figure 5: The WIRE-gadget and the NAND-gadget modified for 7-CPPT

Now we will prove that `6-CPT` is also NP-complete. We are using the same gadgets like for `5-CPPT` but we will pseudo triangulate the "holes" of the graph G. Thus the result will be the fully pseudo-triangulated convex Hull of V.

To prove that `6-CPT` is NP-complete we first make the Graph G_ϕ 3-connected. This will be obtained by adding additional clauses, which connect all the variables and which are trivially true (see Figure 6). Every 3-connected graph admits a convex drawing on a polynomial grid [8]. This embedding can be computed in polynomial time. We use a convex embedding of G_ϕ to make G convex as well. The non-convex gadget chains will be treated like it is shown in Figure 7 (the degree bound will not be violated by this construction). Notice that all non-pseudo-triangular faces of G are convex and have maximal vertex degree 4. We triangulate all those faces in a zig-zag manner, what leads to a degree bound of 6. The NP-completeness of `6-CPT` follows.

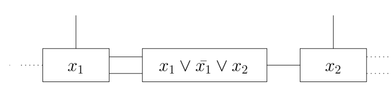


Figure 6: Making G_ϕ 3-connected

Notice that the pseudo-triangulation of G will not be pointed since G is not pointed (NAND-gadget). But the problem remains NP-complete if we restrict ourselves to pointed pseudo-triangulations.

Theorem 2 *For any $k \geq 5$ `k-CPPPT` is NP-complete.*

Proof. To prove NP-completeness we using the same reduction and gadgets, except the NAND gadget, which

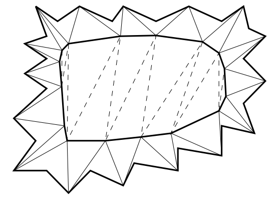
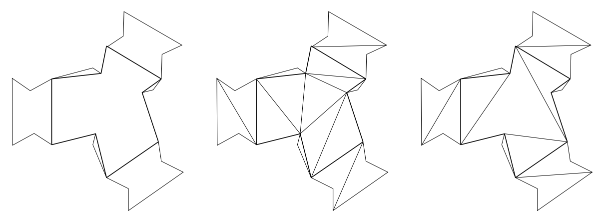


Figure 7: Pseudo-triangulation of the holes induced by G

is the only one where non-pointedness appears. Figure 8.a shows the modified NAND-gadget. Check that it is not possible to pseudo-triangulate the gadget without increasing the vertex degree of some "input vertex" by 2. Otherwise the pointedness and the degree condition would be violated (Figure 8.b). If a vertex is allowed to have two outgoing edges, we can always find a valid pointed pseudo-triangulation. Hence it is always possible to complete the pseudo-triangulation if not all incoming diagonals of the wires are turned right. This gives us a working NAND-gadget for `5-CPPPT` with restriction to pointed pseudo-triangulations. For



(a) (b) (c)

Figure 8: The modified NAND-gadget (a) with a non pointed (b) and pointed (c) pseudo-triangulation.

any $k > 5$ the gadgets will be modified like discussed above. □

If the parameter k would be part of the input, all problems would of course be NP-complete too. Our results showed that the hardness does not depend on the vertex bound given by k. A generalization from `5-CPPPT` to a version where G doesn't have to be polygonal can be made by taking the ideas used for `6-CPPT`.

3 Pseudo-triangulations with low vertex and face degree

In [4] it was shown that it is not possible to generate pseudo-triangulations with low vertex degree *and* low face degree in general. In particular the product of face and vertex degree is in $\Theta(n)$. We present a characterization for point sets which allowing better results. Let $P = \{p_1, \dots, p_n\}$ be a 2-dimensional point set. L_i denotes the i-th *convex layer* of P which is defined as the convex hull of $P \setminus (\cup_{k<i} L_k)$.

Theorem 3 *The point set P can be pseudo-triangulated with vertex degree at most 8 and face degree at most l, where $l = \max\{|L_i|\}$*

Proof. To prove the theorem we show a construction which generates a pseudo-triangulation with the desired properties. As a first step we connect all vertices of each layer in cyclic order. It is left to show that the space between two layers can be pseudo-triangulated appropriately. Without loss of generality we assume that all vertex sets are enumerated in cyclic order and an index i is always understood as (i mod n).

The layer L_i is a convex set and contains the layer L_{i+1}. We denote with L'_i the minimal subset of L_i which still contains L_{i+1} in its convex hull. The edges of L'_i will be connected in cyclic order. The zones between L'_i and L_i are convex polygons. They will be triangulated in a zig-zag manner starting from the cyclicly rightmost corner (see Figure 9.a).

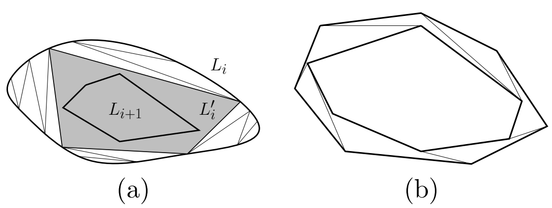


Figure 9: Pseudo-triangulation between L'_i and L_i (a) and between L'_i and L_{i+1} (b).

It is left to describe the pseudo-triangulation between L'_i and L_{i+1}. The points p_i and p_{i+1} of L_{i+1} define the half-plane H_i which partially covers the space outside the polygon defined by L_{i+1}. Let Z_i be the space of $H_i \setminus H_{i+1}$ (Figure 10). All points of $Z_i \cap L'_i$ will be connected with p_{i+1}. This forms a valid pointed pseudo-triangulation. Assume that there are more than 2 points inside $Z_i \cap L'_i$. Then the set L'_i would not be minimal. Thus the vertex of L_{i+1} has at most two outgoing edges to vertices of the set L'_i.

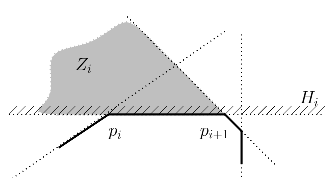


Figure 10: Definition of Z_i

Counting the degree of a vertex we come up with: 2 edges from the layer itself, 2 edges from the layer outside, 2 edges connecting L'_i, one edge to connect L_{i+1} and one edge for the zig-zag triangulation. That leads to a degree bound of 8. The face degree is obviously given by the size of the largest layer. Here all but 2 vertices can be part of one concave chain. Together with the two remaining edges of the pseudo-triangle we receive the face bound stated in the theorem. □

4 Open problems

It would be nice to have some NP-completeness results for unconstrained pseudo-triangulations. Here it would be sufficient to decide, if a given point set allows a degree bound of 5 or 4, since there are only a few point sets with degree bound 3. Another interesting question is, if one could pseudo-triangulate a point set with degree 4, if Steiner points are allowed. How many of these additional points are necessary?

References

[1] O. Aichholzer, M. Hoffmann, B. Speckmann, and C. D. Tóth. Degree bounds for constrained pseudo-triangulations. In *Proc. 15th Canad. Conf. Comput. Geom.*, pages 155–158, 2003.

[2] B. Chazelle, H. Edelsbrunner, M. Grigni, L. J. Guibas, J. Hershberger, M. Sharir, and J. Snoeyink. Ray shooting in polygons using geodesic triangulations. *Algorithmica*, 12:54–68, 1994.

[3] K. Jansen. One strike against the min-max degree triangulation problem. *Comput. Geom. Theory Appl.*, 3:107–120, 1993.

[4] L. Kettner, D. Kirkpatrick, A. Mantler, J. Snoeyink, B. Speckmann, and F. Takeuchi. Tight degree bounds for pseudo-triangulations of points. *Computational Geometry: Theory and Applications*, 25(1&2):1–12, 2003.

[5] D. Kirkpatrick, J. Snoeyink, and B. Speckmann. Kinetic collision detection for simple polygons. In *Proc. 16th Annu. ACM Sympos. Comput. Geom.*, pages 322–330, 2000.

[6] D. Lichtenstein. Planar formulae and their uses. *SIAM J. Comput.*, 11(2):329–343, 1982.

[7] M. Pocchiola and G. Vegter. Computing the visibility graph via pseudo-triangulations. In *Proc. 11th Annu. ACM Sympos. Comput. Geom.*, pages 248–257, 1995.

[8] G. Rote. Strictly convex drawings of planar graphs. In *Proceedings of the 16th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA), Vancouver*, 2005.