

# Polygon triangulation

In computational geometry, **polygon triangulation** is the decomposition of a polygonal area (simple polygon) **P** into a set of triangles,<sup>[1]</sup> i.e., finding a set of triangles with pairwise non-intersecting interiors whose union is **P**.

Triangulations may be viewed as special cases of planar straight-line graphs. When there are no holes or added points, triangulations form maximal outerplanar graphs.

## Contents

- 1 Polygon triangulation without extra vertices**
  - 1.1 Special cases
  - 1.2 Ear clipping method
  - 1.3 Using monotone polygons
  - 1.4 Dual graph of a triangulation
  - 1.5 Computational complexity
- 2 Related problems**
  - 2.1 See also
- 3 References**
- 4 External links**

## Polygon triangulation without extra vertices

Over time a number of algorithms have been proposed to triangulate a polygon.

### Special cases

It is trivial to triangulate any convex polygon in linear time into a fan triangulation, by adding diagonals from one vertex to all other vertices.

The total number of ways to triangulate a convex  $n$ -gon by non-intersecting diagonals is the  $(n - 1)$ -th Catalan number, which equals  $\frac{n \cdot (n+1) \cdots (2n-4)}{(n-2)!}$ , a solution found by Leonhard Euler.<sup>[2]</sup>

A monotone polygon can be triangulated in linear time with either the algorithm of A. Fournier and D.Y. Montuno,<sup>[3]</sup> or the algorithm of Godfried Toussaint.<sup>[4]</sup>

### Ear clipping method

One way to triangulate a simple polygon is based on the two ears theorem, the fact that any simple polygon with at least 4 vertices without holes has at least two 'ears', which are triangles with two sides being the edges of the polygon and the third one completely inside it.<sup>[5]</sup> The algorithm then consists of finding such an ear, removing it from the polygon (which results in a new polygon that still meets the conditions) and repeating until there is only one triangle left.

This algorithm is easy to implement, but slower than some other algorithms, and it only works on polygons without holes. An implementation that keeps separate lists of convex and concave vertices will run in  $O(n^2)$  time. This method is known as *ear clipping* and sometimes *ear trimming*. An efficient algorithm for cutting off ears was discovered by Hossam ElGindy, Hazel Everett, and Godfried Toussaint.<sup>[6]</sup>

## Using monotone polygons

A simple polygon may be decomposed into monotone polygons as follows.<sup>[1]</sup>

For each point, check if the neighboring points are both on the same side of the 'sweep line', a horizontal or vertical line on which the point being iterated lies. If they are, check the next sweep line on the other side. Break the polygon on the line between the original point and one of the points on this one.

Note that if you are moving downwards, the points where both of the vertices are below the sweep line are 'split points'. They mark a split in the polygon. From there you have to consider both sides separately.

Using this algorithm to triangulate a simple polygon takes  $O(n \log n)$  time.

## Dual graph of a triangulation

A useful graph that is often associated with a triangulation of a polygon  $P$  is the dual graph. Given a triangulation  $T_P$  of  $P$ , one defines the graph  $G(T_P)$  as the graph whose vertex set are the triangles of  $T_P$ , two vertices (triangles) being adjacent if and only if they share a diagonal. It is easy to observe that  $G(T_P)$  is a tree with maximum degree 3.

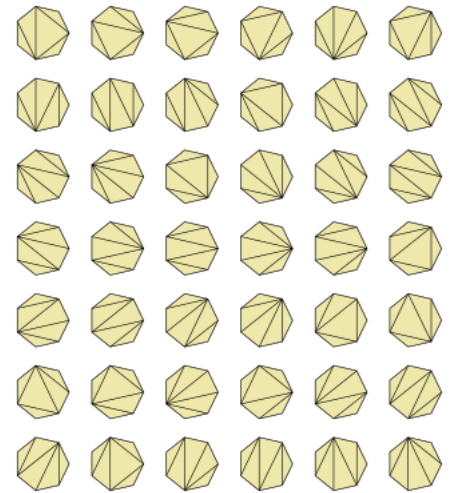
## Computational complexity

For a long time, there was an open problem in computational geometry whether a simple polygon can be triangulated faster than  $O(n \log n)$  time.<sup>[1]</sup> Then, Tarjan & Van Wyk (1988) discovered an  $O(n \log \log n)$ -time algorithm for triangulation,<sup>[7]</sup> later simplified by Kirkpatrick, Klawe & Tarjan (1992).<sup>[8]</sup> Several improved methods with complexity  $O(n \log^* n)$  (in practice, indistinguishable from linear time) followed.<sup>[9][10][11]</sup>

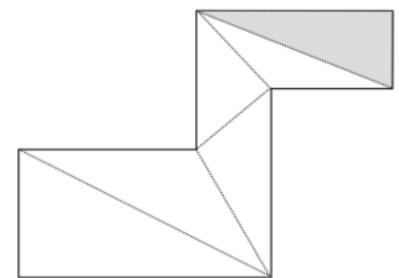
Bernard Chazelle showed in 1991 that any simple polygon can be triangulated in linear time, though the proposed algorithm is very complex.<sup>[12]</sup> A simpler randomized algorithm with linear expected time is also known.<sup>[13]</sup>

Seidel's decomposition algorithm and Chazelle's triangulation method are discussed in detail in Li & Klette (2011).<sup>[14]</sup>

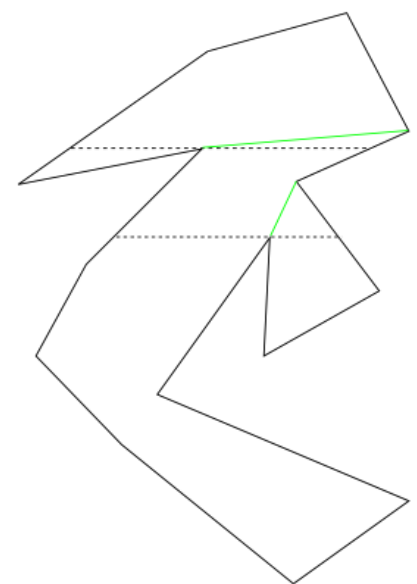
The time complexity of triangulation of an  $n$ -vertex polygon *with* holes has an  $\Omega(n \log n)$  lower bound.<sup>[1]</sup>



The 42 possible triangulations for a convex heptagon (7-sided convex polygon). This number is given by the 5th Catalan number.



A polygon ear



Breaking a polygon into monotone polygons

## Related problems

Minimum-weight triangulation is a triangulation in which the goal is to minimize the total edge length, rather than the number of triangles.

Both triangulation problems are a special case of triangulation (geometry) and a special case of polygon partition.

A point set triangulation is a polygon triangulation of the convex hull of a set of points. A Delaunay triangulation is another way to create a triangulation based on a set of points.

A related problem is polygon triangle covering, in which the triangles may overlap. Another related problem is tiling by polygons, where the goal is to cover the entire plane with polygons of pre-specified shapes.

### See also

- Nonzero-rule
- Catalan number
- Planar graph

## References

- Mark de Berg, Marc van Kreveld, Mark Overmars, and Otfried Schwarzkopf (2000), *Computational Geometry* (2nd revised ed.), Springer-Verlag, ISBN 3-540-65620-0 Chapter 3: Polygon Triangulation: pp.45–61.
- Pickover, Clifford A., *The Math Book*, Sterling, 2009: p. 184.
- Fournier, A.; Montuno, D. Y. (1984), "Triangulating simple polygons and equivalent problems", *ACM Transactions on Graphics*, **3** (2): 153–174, ISSN 0730-0301 (<https://www.worldcat.org/issn/0730-0301>), doi:10.1145/357337.357341 (<https://doi.org/10.1145/357337.357341>)
- Toussaint, Godfried T. (1984), "A new linear algorithm for triangulating monotone polygons," *Pattern Recognition Letters*, **2** (March):155–158.
- Meisters, G. H., "Polygons have ears." *American Mathematical Monthly* **82** (1975). 648–651
- ElGindy, H.; Everett, H.; Toussaint, G. T. (1993). "Slicing an ear using prune-and-search". *Pattern Recognition Letters*. **14** (9): 719–722. doi:10.1016/0167-8655(93)90141-y ([https://doi.org/10.1016/0167-8655\(93\)90141-y](https://doi.org/10.1016/0167-8655(93)90141-y)).
- Tarjan, Robert E.; Van Wyk, Christopher J. (1988), "An  $O(n \log \log n)$ -time algorithm for triangulating a simple polygon", *SIAM Journal on Computing*, **17** (1): 143–178, MR 925194 (<https://www.ams.org/mathscinet-getitem?mr=925194>), doi:10.1137/0217010 (<https://doi.org/10.1137/0217010>).
- Kirkpatrick, David G.; Klawe, Maria M.; Tarjan, Robert E. (1992), "Polygon triangulation in  $O(n \log \log n)$  time with simple data structures", *Discrete and Computational Geometry*, **7** (4): 329–346, MR 1148949 (<https://www.ams.org/mathscinet-getitem?mr=1148949>), doi:10.1007/BF02187846 (<https://doi.org/10.1007/BF02187846>).
- Clarkson, Kenneth L.; Tarjan, Robert; van Wyk, Christopher J. (1989), "A fast Las Vegas algorithm for triangulating a simple polygon", *Discrete and Computational Geometry*, **4**: 423–432, doi:10.1007/BF02187741 (<https://doi.org/10.1007/BF02187741>).
- Seidel, Raimund (1991), "A Simple and Fast Incremental Randomized Algorithm for Computing Trapezoidal Decompositions and for Triangulating Polygons", *Computational Geometry: Theory and Applications*, **1**: 51–64, doi:10.1016/0925-7721(91)90012-4 ([https://doi.org/10.1016/0925-7721\(91\)90012-4](https://doi.org/10.1016/0925-7721(91)90012-4))
- Clarkson, Kenneth L.; Cole, Richard; Tarjan, Robert E. (1992), "Randomized parallel algorithms for trapezoidal diagrams", *International Journal of Computational Geometry & Applications*, **2** (2): 117–133, MR 1168952 (<https://www.ams.org/mathscinet-getitem?mr=1168952>), doi:10.1142/S0218195992000081 (<https://doi.org/10.1142/S0218195992000081>).

12. Chazelle, Bernard (1991), "Triangulating a Simple Polygon in Linear Time", *Discrete & Computational Geometry*, **6**: 485–524, ISSN 0179-5376 (<https://www.worldcat.org/issn/0179-5376>), doi:10.1007/BF02574703 (<https://doi.org/10.1007%2FBF02574703>)
13. Amato, Nancy M.; Goodrich, Michael T.; Ramos, Edgar A. (2001), "A Randomized Algorithm for Triangulating a Simple Polygon in Linear Time" ([http://parasol.tamu.edu/publications/abstract.php?pub\\_id=185](http://parasol.tamu.edu/publications/abstract.php?pub_id=185)), *Discrete & Computational Geometry*, **26** (2): 245–265, ISSN 0179-5376 (<https://www.worldcat.org/issn/0179-5376>), doi:10.1007/s00454-001-0027-x (<https://doi.org/10.1007%2Fs00454-001-0027-x>)
14. Li, Fajie; Klette, Reinhard (2011), *Euclidean Shortest Paths*, Springer, ISBN 978-1-4471-2255-5, doi:10.1007/978-1-4471-2256-2 (<https://doi.org/10.1007%2F978-1-4471-2256-2>).

## External links

---

- [Demo as Flash swf \(http://computacion.cs.cinvestav.mx/~anzures/geom/triangulation.php\)](http://computacion.cs.cinvestav.mx/~anzures/geom/triangulation.php), A Sweep Line algorithm.
  - [Song Ho's explanation of the OpenGL GLU tessellator \(http://www.songho.ca/opengl/gl\\_tessellation.html\)](http://www.songho.ca/opengl/gl_tessellation.html)
- 

Retrieved from "[https://en.wikipedia.org/w/index.php?title=Polygon\\_triangulation&oldid=807080153](https://en.wikipedia.org/w/index.php?title=Polygon_triangulation&oldid=807080153)"

---

**This page was last edited on 25 October 2017, at 19:37.**

Text is available under the [Creative Commons Attribution-ShareAlike License](#); additional terms may apply. By using this site, you agree to the [Terms of Use](#) and [Privacy Policy](#). Wikipedia® is a registered trademark of the [Wikimedia Foundation, Inc.](#), a non-profit organization.