# Point Triangulation using Graham's Scan 

Vasyl Tereshchenko, Yaroslav Tereshchenko<br>Taras Shevchenko National University of Kyiv<br>Kyiv, Ukraine<br>vtereshch@gmail.com, y_ter@ukr.net


#### Abstract

In this paper, we propose a triangulation method for a set of points in the plane. The method is based on the idea of constructing convex layers by Graham's scan. It allows to develop an algorithm with the optimal complexity of O(NlogN) and an easy implementation. First, convex hulls are constructed for the set $S$ of $N$ points, forming $k$ layers. Then, each layer is triangulated in one scan of the adjacent convex hulls. Algorithm is easily parallelized: each layer can be triangulated independently. The main feature of the proposed algorithm is that it has a very simple implementation and the elements (triangles) of the resulting triangulation are presented in the form of simple and at the same time fast data structures: concatenable triangle queue or triangle tree. This makes the algorithm convenient for solving a wide range of applied problems of computational geometry and computer graphics, including simulation in science and engineering, rendering and morphing.


Keywords: Triangulation, Convex hull, Set of points, Convex layers, Graham's scan
Received: 28 June 2014, Revised 2 August 2014, Accepted 10 August 2014
© 2014 DLINE. All Rights Reserved

## 1. Introduction

This paper proposes an optimal triangulation algorithm for a set of points in the plane. The main advantage of triangulation is that from an object, which is potentially very complex, we can move on to more simple polygons (triangles) for their further study.

Relevance. Today, there exist a number of efficient algorithms for solving the problem of triangulation of a set of points [1]. The following groups of triangulation algorithms are distinguished: iterative algorithms (least efficient and quite difficult to implement) [2], algorithms, based on the «divide-and-conquer» strategy (the fastest and relatively easy to implement) [2-5], direct construction algorithms (have good (even linear) average construction time, easy to implement) [6] and two-pass algorithms (most difficult to implement, not very effective) [7].

In my opinion, most effective are triangulation methods, based on the «divide-and-conquer» strategy, that have time complexity $O(N \log N)$ in worst and average cases [2-5]. Among them, algorithms that use concatenable queue data structure at the merge step can be singled out $[4,5]$. These algorithms give optimal results in terms of computational complexity $-\theta(N \log N)$, but it is desirable to have a simpler implementation. So naturally, the question arises - is it possible to develop an algorithm that would

$100 \quad$ Journal of Data Processing |  | Volume | 4 | Number | 3 | September 2014 |
| :--- | :--- | :--- | :--- | :--- | :--- | :--- |

give high efficiency and at the same time would be simple to implement? Once again, we stress that this is important in terms of practical use of the algorithm. An example of such method is Graham's algorithm for construction of convex hulls, that has optimal computation complexity and at the same time is very easy to implement [8]. This method has inspired to develop a triangulation algorithm as efficient and simple in implementation as the algorithm for convex hull construction.

Goal. Develop a triangulation algorithm based on the Graham's method, which would have optimal time complexity $(\theta N \log N)$ ) and at the same time would be easy to implement.

Originality. A new algorithm for triangulation of a set of points is proposed in the paper. The new algorithm is based on the Graham's method and has an optimal complexity of $\theta(N \log N)$.

## 2. Problem and Solution

Problem. Triangulate the set $S$ of $N$ points in the plane using Graham's scan with the time complexity of $\theta(N \log N)$.

### 2.1 Method of Solving the Problem

Let $S$ be the set of $N$ points in the plane, Figure 1.


Figure 1. Example of an input set of points

1. According to the Graham's method, we find centroid $q\left(x_{q}, y_{q}\right)$ of the first three noncollinear points. For example, in figure 1 it is the centroid for points $P_{1}, P_{2}, P_{3}$ with coordinates $x_{q}, y_{q}$, accordingly Figure 2 :

$$
x_{q}=\frac{x_{1}+x_{2}+x_{3}}{3}, y_{q}=\frac{y_{1}+y_{2}+y_{3}}{3} .
$$



Figure 2. Definition of point $q$
2. We sort the given set S of N points by their polar angle (counterclockwise), getting an ordered list $U$. For our example from figure 1 , we will get $U=\left\{P_{4}, P_{5}, P_{6}, P_{7}, P_{12}, P_{8}, P_{11}, P_{3}, P_{10}, P_{9}, P_{2}, P_{1}\right\}$.
3. We use the Graham's scan for the list $U$, as a result obtaining the convex hull for the set S with the boundary $C H(S)=\left\{P_{5}, P_{12}\right.$, $\left.P_{11}, P_{10}, P_{9}, P_{2}, P_{1}\right\}$, Figure 3 .


Figure 3. Convex hull for $S$
4. For the set of points $S_{1}$, remaining inside the convex hull, we construct convex hull $\mathrm{CH}\left(S_{1}\right)$ using Graham's scan. Similarly, we construct convex hulls $C H\left(S_{2}\right), \ldots, C H\left(S_{k}\right)$ for the following sets $S_{2}, \ldots, S_{\mathrm{k}}$, until it is possible, Figure 4.


Figure4. Constructing convex hulls
5. Accordingly we triangulate layers, which are formed by the adjacent convex hull boundaries, Figure. 5. This can be done even during each subsequent scan, and in case of possibility of parallel processing, each layer can be triangulated independently.


Figure 5. Triangulating of convex layers

1 | 02 | Journal of Data Processing | Volume | 4 | Number | 3 | September |
| :--- | :--- | :--- | :--- | :--- | :--- | :--- |
| 2014 |  |  |  |  |  |  |

## 3. Data Structure Construction

The question arises, how to present the resulting triangulation in the form of a certain data structure that could be used for processing and solving the following problems: coloring, finding intersections, morphing, rendering, Boolean operations etc. In this case, the triangulation procedure provides a convenient way for its maintenance, namely:

1) During the scan of construction of every new triangle, we assign it a name and add it to the created, cyclically ordered list of such triangles. Figure. 6 shows an appropriate example.


Figure 6. Construction of the list of triangles: $"=\{1,2,3,4,5,6,7,8,9,10,11,12,13,14,15\}$
2) The formed list can be presented in the form of a concatenable queue (Figure. 7), which maintains connectivity in each layer (blue and brown lines) and contains pointers to the links between the edges of the adjacent layers (green lines).

This allows to carry out logarithmic search for the triangulation on anv laver and in anv direction.


Figure 7. Data structure in the form of a linked queue for fig. 6 . First layer edges $\{1,2,3,4\}$ are marked in red, second layer edges $\{5,6,7,8,9,10,11,12,13,14,15\}$ are marked in blue
3) The actual layered triangulation can be presented in the form of a binary edge tree (Figure 8), which maintains connectivity.

Using this data structure, logarithmic search can be carried out from any triangle.

$$
\begin{array}{llllllll}
\hline \text { Journal of Data Processing } & \text { Volume } & 4 & \text { Number } & 3 & \text { September } & 2014 & 103 \\
\hline
\end{array}
$$

## 4. Complexity

Theorem. The proposed algorithm has the same time complexity as Graham's method for convex hull construction $-O(N \log N)$ and uses linear space.

Proof. $1^{\text {st }}, 2^{\text {nd }}$ and $3^{\text {rd }}$ steps are steps of the Graham's algorithm and they require $O(N), O(N \log N), O(N)$ time respectively. $4^{\text {th }}$ step is the Graham's scan and it requires $O(N)$ time, but for the set of points, left after step $3.5^{\text {th }}$ step - layer triangulation, which is carried out (considering the ordering of points at the border of each convex hull by their polar angle) in one scan using $O(N)$ time.


Figure 8. Data structure in the form of a binary edge tree

## 5. Conclusions

An optimal method for triangulation of a set of points with the complexity of $O(N \log N)$ is proposed in the paper. This algorithm is based on the Graham's scan for computing the convex hull. First, convex hulls are constructed for the set S of N points, forming $k$ layers. Then, each layer is triangulated in one scan of the adjacent convex hulls. Algorithm is easily parallelized: each layer can be triangulated independently. The main feature of the proposed algorithm is that it has a very simple implementation. The algorithm has application to solving a wide range of applied problems of simulation in science and engineering.

The feature of the proposed method is not only simple process of triangulation constructing, but also convenient representation of its elements for the further use in the form of data structures. These data structures are concatenable queue or a binary faces tree.

## References

[1] Skvorcov .V. Delaunay Triangulation and Its Application. Izd Tomsk, Gos. Univ., Tomsk, 2002.
[2] Skvorcov., Y.I. Kostiuk. Effectivnie algoritmi postroenia trianguliacii Delone. Geoinformatika. Teoria i praktika.- Tomsk: Un-t., 1998, P. 22-47.
[3] Gubias L., Stolfi J., Primitives for the manipulation of general subdivisions and the computation of Voronoi diagrams . ACM Transactions on Graphics.- 4, N 2. - 1985, P. 74-123.
[4] Tereshchenko V. The generalized approach of the decision of some problems of computing geometry on the basis of recursively-parallel technology. Naukovi notatki, Luck-2008. - 22 , N 2.- , 2008, P. 344-349.
[5] Tereshchenko V.N., Anisimov A.V. Recursion and parallel algorithms in geometric modeling problems. Journal: Cybernetics and Systems Analysis:- 46, N 2.- 2010, P. 173-184.
[6] Kirpatric D.G. Optimal search in planar subdivisions. SIAM J. Comput. - 12, N 1. -1983, P. 28-35.
[7] Lewis B., Robinson J. Triangulation of planar regions with applications . The Computer Journal. - 21, N 4. - 1978, P. 324-332.

| 104 | Journal of Data Processing | Volume | 4 | Number | 3 | September | 2014 |
| :--- | :--- | :--- | :--- | :--- | :--- | :--- | :--- |

[8] Graham, R. L. (1972). An efficient algorithm for determining the convex hull of a finite planar set. Info. Proc. Lett.- 1.- , p. 132133.


Vasyl Tereshchenko. Professor, Doctor of Phys. - Math. Science, Taras Shevchenko National University of Kyiv, Faculty of Cybernetics. In 1986 graduated from the Mathematics and Mechanics Faculty of Kyiv National Taras Shevchenko University. In 1992 graduated from graduate school and in 1993 defended PhD thesis on the degree C.Sci. (Phys-Math.). Dissertation theme - Nonstationary problems of thermo elasticity for piecewise-homogeneous bodies. In 2011 defended a dissertation for the degree of doctor of Phys. - Math. Science. Dissertation theme - Constructing the common algorithmic environment for solving a set of problems of computational geometry. 01.05.01 - Theoretical bases of computer science and cybernetics. In 1994 - Associate Professor of Faculty of Cybernetics KNTSU. Since 2013 - Full Professor of Faculty of Cybernetics KNTSU. Lecturer in computer graphics and in computational geometry, and also in databases and in the theory of algorithms.

Yaroslav Tereshchenko Since 2011 student of the Faculty of Cybernetics, Kyiv National Taras Shevchenko University. Deals with computational geometry and software engineering.

