Main page
Contents
Featured content
Current events
Random article
Donate to Wikipedia
Wikipedia store

Interaction
Help
About Wikipedia
Community portal
Recent changes
Contact page

Tools
What links here
Related changes
Upload file
Special pages
Permanent link
Page information
Wikidata item
Cite this page

Print/export
Create a book
Download as PDF
Printable version

Languages
Add links

Jump-and-Walk algorithm

From Wikipedia, the free encyclopedia

Jump-and-Walk is an algorithm for point location in triangulations (though most of the theoretical analysis were performed in 2D and 3D random Delaunay triangulations). Surprisingly, the algorithm does not need any preprocessing or complex data structures except some simple representation of the triangulation itself. The predecessor of Jump-and-Walk was due to Lawson (1977) and Green and Sibson (1978), which picks a random starting point S and then walks from S toward the query point Q one triangle at a time. But no theoretical analysis was known for these predecessors until after mid-1990s.

Jump-and-Walk picks a small group of sample points and starts the walk from the sample point which is the closest to Q until the simplex containing Q is found. The algorithm was a folklore in practice for some time, and the formal presentation of the algorithm and the analysis of its performance on 2D random Delaunay triangulation was done by Devroye, Mucke and Zhu in mid-1990s (the paper appeared in Algorithmica, 1998). The analysis on 3D random Delaunay triangulation was done by Mucke, Saias and Zhu (ACM Symposium of Computational Geometry, 1996). In both cases, a boundary condition was assumed, namely, Q must be slightly away from the boundary of the convex domain where the vertices of the random Delaunay triangulation are drawn. In 2004, Devroye, Lemaire and Moreau showed that in 2D the boundary condition can be withdrawn (the paper appeared in Computational Geometry: Theory and Applications, 2004).

Jump-and-Walk has been used in many famous software packages, e.g., QHULL, Triangle and CGAL.

References [edit]

P. Green and Sibson. Computing Dirichlet tessellations in the plane. The Computer Journal, 21:168-173, 1978.

C. Lawson. Software for C1 surface interpolation. In J.R. Rice, editor, Mathematical Software III, pages 161-194, 1977 (Academic Press, NY).

L. Devroye, C. Lemaire and J-M Moreau. Expected time analysis for Delaunay point location. Computational Geometry: Theory and Applications, 29:61-89, 2004.

L. Devroye, E. Mucke and B. Zhu. A note on point location in Delaunay triangulations of random points. Algorithmica, 22:477-482, 1998.

E. Mucke, I. Saias and B. Zhu. Fast randomized point location without preprocessing in two- and three-dimensional Delaunay triangulations. Proc. 12th ACM Symposium on Computational Geometry, Pages 274-283, 1996.

Categories: Triangulation (geometry) | Algorithms

This page was last modified on 8 March 2014, at 18:47.