Introduction to the Computational Geometry Algorithms Library CGAL

CGAL

www.cgal.org

Monique Teillaud

INRIA
Part I

The CGAL Open Source Project
Goals

• Promote the research in Computational Geometry (CG)

• “make the large body of geometric algorithms developed in the field of CG available for industrial applications”

⇒ robust programs
History

- Development started in 1995
History

- Development started in 1995
- January, 2003: creation of **Geometry Factory** (INRIA startup)
- November, 2003: **Open Source Project**
- March 2012: Release 4.0
Distribution

- release cycle of 6 months
Distribution

- release cycle of 6 months
- from the INRIA gforge
  - > 1,000 downloads per month
- included in Linux distributions (Debian, etc)
- available through macport

CGAL triangulations, meshes, etc, can be used in Java or Python.
Distribution

- release cycle of 6 months

- from the INRIA gforge
  - > 1,000 downloads per month
- included in Linux distributions (Debian, etc)
- available through macport

- triangulation packages integrated in Matlab

- New CGAL-bindings (implemented with SWIG)
  - CGAL triangulations, meshes, etc, can be used in Java or Python
License

- a few basic packages under LGPL
- all other packages under GPLv3+

- **commercial** licenses through GEOMETRY FACTORY
CGAL welcomes new contributions

• Registration
  Preliminary information

• Submission of the specifications (and design)
  Review by the Editorial Board

• Integration
  Test-suite running on all supported platforms

quality → helps authors get credit for their work.

Information on the CGAL web site
“Getting Involved in the CGAL Project”
Contributors keep their identity

- Listed as **authors** in the manual

**3D Triangulations**
Sylvain Pion and Monique Teillaud

This package allows to build and handle triangulations for point sets in three dimensions. Any CGAL triangulation covers the convex hull of its vertices. Triangulations are build incrementally and can be modified by insertion or removal of vertices. They offer point location facilities.

The package provides plain triangulation (whose faces depends on the insertion order of the vertices) and Delaunay triangulations. Regular triangulations are also provided for sets of weighted points. Delaunay and regular triangulations offer nearest neighbor queries and primitives to build the dual Voronoi and power diagrams.

**3D Triangulation Data Structure**
Sylvain Pion and Monique Teillaud

This package provides a data structure to store a three-dimensional triangulation that has the topology of a three-dimensional sphere. The package acts as a container for the vertices and cells of the triangulation and provides basic combinatorial operations on the triangulation.

**3D Periodic Triangulations**
Manuel Caroli and Monique Teillaud

This package allows to build and handle triangulations of point sets in the three dimensional flat torus. Triangulations are build incrementally and can be modified by insertion or removal of vertices. They offer point location facilities.

The package provides Delaunay triangulations and offers nearest neighbor queries and primitives to build the dual Voronoi diagrams.

- Mentioned on the “People” web page

- **Copyright** kept by the institution of the authors
Editorial Board

- Responsible for the **quality** of CGAL
- Decides about technical matters
- Coordinates communication and promotion
- ...
Currently:

Eric Berberich (Max-Planck-Institut für Informatik)
Efi Fogel (Tel Aviv University)
Michael Hemmer (Tel Aviv University)
Bernd Gärtner (ETH Zürich)
Michael Hoffmann (ETH Zürich)
Menelaos Karavelas (University of Crete and FORTH)
Andreas Fabri (GEOMETRY FACTORY)
Sébastien Loriot (GEOMETRY FACTORY)
Laurent Rineau (GEOMETRY FACTORY)
Marc Pouget (INRIA Nancy - Grand Est)
Pierre Alliez (INRIA Sophia Antipolis - Méditerranée)
Monique Teillaud (INRIA Sophia Antipolis - Méditerranée)
Mariette Yvinec (INRIA Sophia Antipolis - Méditerranée)
The CGAL community

- 4,000 subscribers to announcement list (7,000 for gcc)
- 1,000 subscribers to discussion list (600 in gcc-help)
- 50 developers registered on developers list (≈ 25 active)
  - developers wiki
  - 1-2 developers meetings per year, 1 week long
Part II

Contents of CGAL
Numbers

- 500,000 lines of **C++** code
- several platforms
  - Linux, MacOS, Windows
  - g++, VC++
- 3,500 pages manual
  - split into 80 chapters
Part III

Strengths
**Numerical Robustness**

`double` → **inconsistencies** in **predicate** evaluations

Ex: **orientation** of 2D points = sign of a polynomial expression

\[
p = (0.5 + x.u, 0.5 + y.u)
\]
\[
0 \leq x, y < 256, \quad u = 2^{-53}
\]

\[
q = (12, 12)
\]
\[
r = (24, 24)
\]

**orientation**(\(p, q, r\))

evaluated with `double`

\[
(x, y) \mapsto \begin{array}{c}
> 0, \\
= 0, \\
< 0
\end{array}
\]

[Kettner et al.]
Numerical Robustness

Speed and exactness through

**Exact Geometric Computation**

\[ \neq \]

**exact arithmetics**

**Filtering Techniques** (interval arithmetics, etc) 
exact arithmetics only when needed

**Degenerate cases** explicitly handled
Flexibility

Generic Algorithms

typedef
    CGAL::Exact_predicates_inexact_constructions_kernel K;

typedef CGAL::Delaunay_triangulation_2<K> Delaunay;

K provides
• 2D points: \((x, y)\)
• predicates:
    orientation, in_circle
• constructions
typedef CGAL::Exact_predicates_inexact_constructions_kernel K;
typedef CGAL::Projection_traits_xy_3<K> Traits;
typedef CGAL::Delaunay_triangulation_2<Traits> Terrain;

Traits provides
• 3D points: \( (x, y, z) \)
• predicates:
  orientation, in_circle
  on \( x \) and \( y \) coordinates only
• constructions
Efficiency

Ex: 3D Delaunay triangulations

- fully dynamic (also weighted triangulations)
- 1 M points \( \approx \) 10 sec (\( \approx \) 10 \( \mu \)sec /point)
- basis for 3D meshes
- integrated in Matlab 2009

[Pion, T.]
Part IV

Applications
Printed circuits

[Pedro M.M. de Castro, Master internship]

Used: Arrangements, 2D circular kernel
VoxMorph:
3-scale freeform deformation of large voxel grids

[Bloch et al.]

Used: 3D mesher
Modeling of flow on pore scale

Used: 3D mesher
Modeling protein interfaces and protein binding patches

[Cazals et al.]

Used: 3D weighted Delaunay triangulations, 3D alpha-shapes, 3D spherical kernel
Description of protein-RNA interactions

Used: 3D Delaunay triangulations,
3D periodic Delaunay triangulations,
3D alpha-shapes

[Bernauer et al.]
Users in many fields

- **Molecular Modeling**
- Particle Physics, Fluid Dynamics, Microstructures
- Medical Modeling and Biophysics
- Geographic Information Systems
- Games
- Motion Planning
- Sensor Networks
- **Architecture, Buildings Modeling, Urban Modeling**
- **Astronomy**
- 2D and 3D Modelers
- Mesh Generation and Surface Reconstruction
- Geometry Processing
- Computer Vision, Image Processing, Photogrammetry
- Computational Topology and Shape Matching
- **Computer Algebra**
- Computational Geometry and Geometric Computing
- ...
Some Commercial Users
To know more

www.cgal.org

Tutorial at the

Minisymposium on Publicly Available
Geometric/Topological Software

Tuesday — 17:20 - 18:05